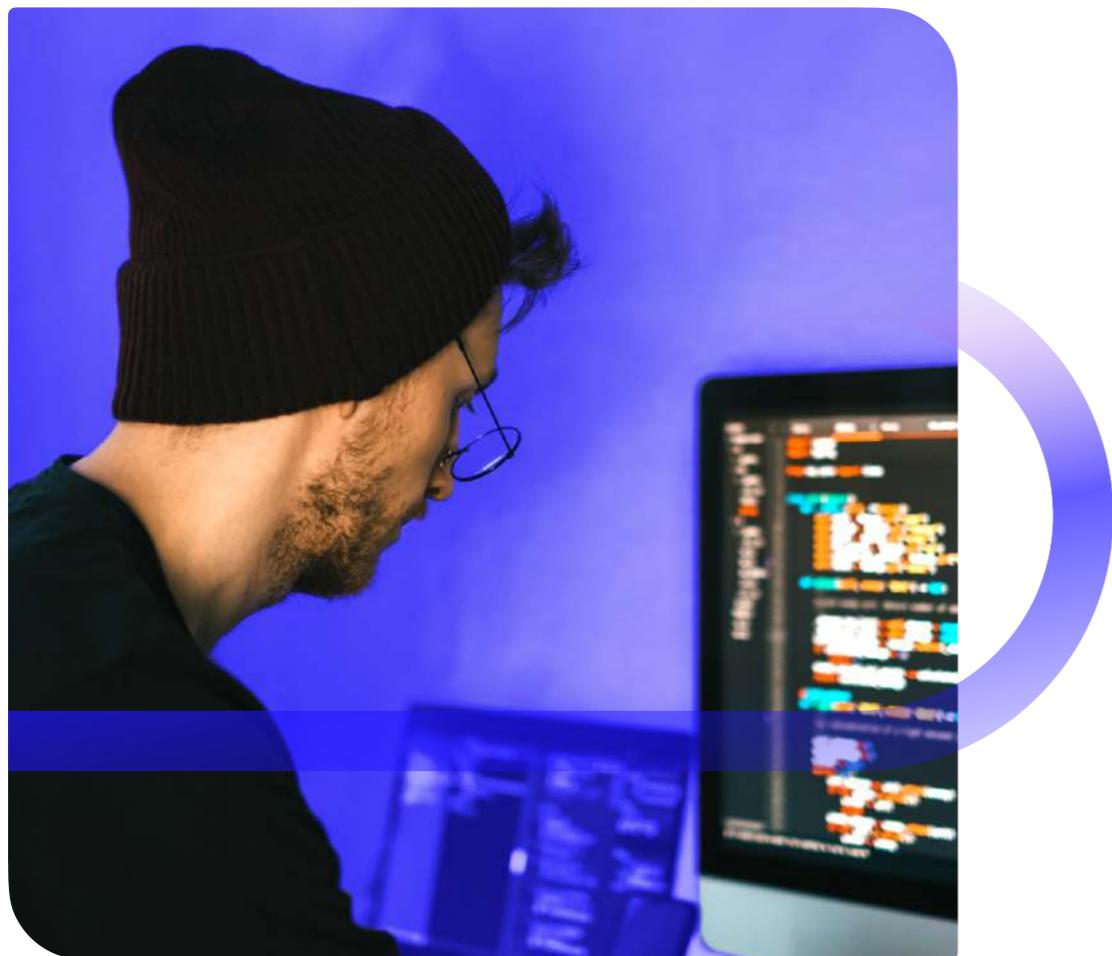




Cloud Developer **Syllabus**

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

Elevate your tech skills with our innovative program focused on Cloud Computing, Microservices, and Serverless Technologies. Master the fundamentals of AWS, learn to build and deploy scalable, full-stack applications on the cloud, and grasp the principles of modern cloud architecture. Delve into microservices, understanding architecture patterns, service replication, and independent scaling. Harness the power of serverless technologies, creating REST APIs and implementing security best practices. Our hands-on exercises offer practical experience, equipping you to design, build, and deploy complex applications. Stay ahead in the digital era with our comprehensive tech program.

Prerequisites

- ✓ Basic Computer Literacy
- ✓ Relational Database Basics
- ✓ Intermediate knowledge of JavaScript programming
- ✓ Learners should be familiar with the use of Git and Github
- ✓ Linux command line basics

Educational Objectives**A GRADUATE OF THIS PROGRAM WILL BE ABLE TO:**

- ✓ Use AWS services to deploy a static website
- ✓ Create a monolithic application on AWS
- ✓ Use CI/CD to break apart a monolithic application into AWS microservices
- ✓ Employ AWS Lambda to build a Serverless application

Program information**LENGTH OF PROGRAM*:**

4 months

**SKILL LEVEL:**

Intermediate

**SCHOOL:**

Cloud Computing

**SOFTWARE/HARDWARE AND VERSION REQUIREMENTS:**

All work for this program may be completed within the Udacity Workspaces. However, if working on a local machine, learners will need the following required dependencies: IDE such as VSCode, AWS CLI, an AWS account, Node.js, NPM, ElasticBeanstalk CLI, Docker, and Iconic CLI.

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.



Supporting Lesson Content

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.

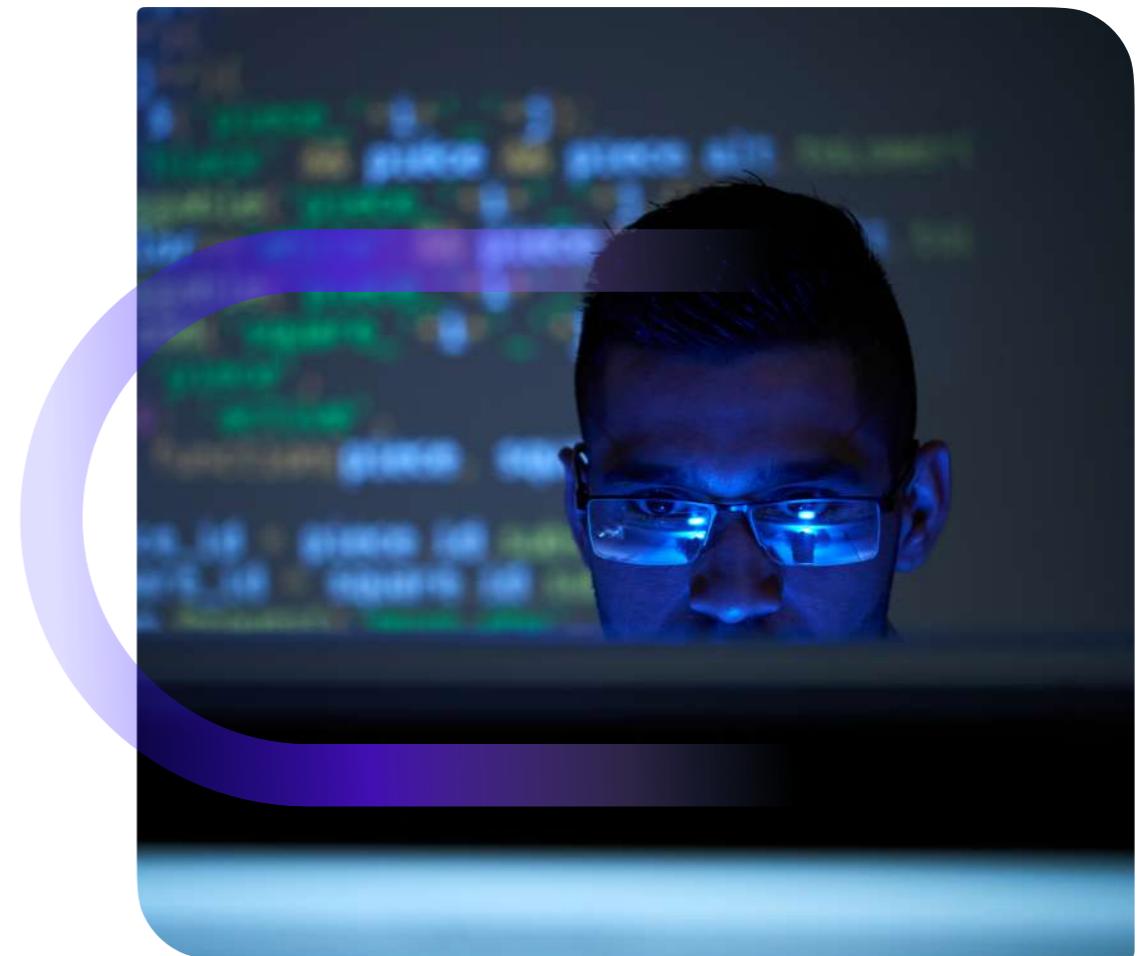
AWS MANAGEMENT

- 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:

Full Stack Apps on AWS

In this course, you will learn how to create and deploy full-stack, REST-based applications on AWS. You'll start by understanding the design paradigm of modern cloud applications, before exploring how to store user data in the cloud using AWS RDS and implement a filestore for media using AWS S3. Learn how to integrate and utilize these data services within your server applications, followed by mastering application deployment using AWS Elastic Beanstalk. The course also equips you with knowledge about common security pitfalls and modern techniques to handle authentication challenges. You'll discover how to ensure the operation and scaling of your services, and maintain your cloud applications efficiently as demand and dependencies evolve.



PROJECT #2

Image Processing Microservice on AWS

In this project, learners will develop a cloud-based application for uploading, listing, and filtering images. They will use Node.js/Express, a popular JavaScript framework for networked application development to develop this application. They will implement a REST API to issue commands using HTTP, store data in Amazon Web Services Relational Data Service (RDS) and S3, extend the codebase with secure authentication sign-on features, and deploy to Amazon Web Services Elastic Beanstalk.



Supporting Lesson Content

INTRODUCTION TO FULL STACK APPS ON AWS

- Identify the components of full-stack applications on the AWS platform.
- Review the tools and environment for building full-stack apps on AWS.



Supporting Lesson Content

REST API DEVELOPMENT

- Learn how communication over the internet works, including the fundamentals of request-response cycles, HTTP methods, and status codes, which form the backbone of REST APIs.
- Design effective APIs, focusing on aspects like routing, data handling, and error management.
- Compare and contrast the pros and cons of monolithic versus loosely coupled (or microservices-based) systems.
- Create REST API endpoints.

DEVELOPING WITH CLOUD DATABASES AND STORAGE

- Provision a cloud database using AWS RDS service. Understand the setup process, configurations, and common management tasks.
- Integrate your application with a cloud database. Learn how to establish a connection, perform database operations, and handle responses in your application.
- Create an S3 bucket and connect your application to AWS S3 for storing and retrieving media files or other data.

DEPLOYING APPLICATIONS ON AWS

- Deploy Applications on Elastic Beanstalk: learn the process of configuring, deploying, and launching your applications to Elastic Beanstalk.
- Deploy your service on Elastic Beanstalk as a Docker container, to enhance portability and consistency.
- Monitor the performance and health of your applications deployed on AWS. Understand the concepts of scalability, how to anticipate needs, and efficiently adjust resources to handle varying demand.

SECURING AWS APPLICATIONS

- Learn best practices for secure password storage, including the use of encryption.
- Implement user authentication mechanisms in the context of AWS applications.
- Use effective strategies for storing secrets (like API keys or other sensitive data) to enhance the security of your application.

AWS FRONTEND DEVELOPMENT

- Understand Frontend Application Development on AWS and the interaction with backend services.
- Deploy a React application to AWS Elastic Beanstalk that interacts with a backend.

COURSE #3:

Monolith to Microservices at Scale

Microservices are becoming the default mode of developing and deploying applications at scale. The microservices architecture makes it easier to scale an application to a large system and is a great enabler for continuous integration and delivery. Microservices architecture allows independent scaling, independent releases and deployments, and independent development so that each service has its own codebase. In this course we will cover the best practices on how to develop and deploy microservices. Learners will focus on topics such as different microservice architecture patterns, independent scaling, resiliency, securing microservices, and best practices for monitoring and logging. By the end of this course, learners should be able to design and build an application using a microservice architecture

PROJECT #3:

Refactor Monolith to Microservices and Deploy

In this project, learners will take an existing application named Udagram and refactor it into a microservice architecture with lean services. Build out a CI/CD process that automatically builds and deploys Docker images to a Kubernetes cluster. The Kubernetes cluster will be configured to help solve common challenges related to scale and security.



Supporting Lesson Content

MICROSERVICES DESIGN PRINCIPLES & BEST PRACTICES

- Learn different microservices architecture designs.
- Learn how to divide an application into microservices.

CONTAINERS USING DOCKER

- Build and run your first container image using Docker.
- Debug container and store these images using container registry.

AUTOMATING THE APPLICATION DEVELOPMENT LIFECYCLE

- Understand CI/CD benefits and use Travis to build CI/CD pipeline.
- Integrate GitHub and CI/CD and automate testing with CI.

ORCHESTRATION WITH KUBERNETES

- Learn the fundamentals of Kubernetes.
- Configure and launch an auto-scaling, self-healing Kubernetes cluster.
- Deploy your microservices using Kubernetes cluster

BEST PRACTICES & DESIGN PATTERNS FOR KUBERNETES IN PRODUCTION

- Implement service registration and discovery.
- Configure scaling and self-healing.
- Secure microservices.
- Implement monitoring and logging for microservices deployment.
- Improve resilience and availability into cloud applications.

COURSE #4:

Develop and Deploy Serverless Apps

Serverless technologies have become popular because they can increase the speed of development and drastically reduce the cost of running a cloud infrastructure. This course combines the theory of using serverless technologies with the practice of developing a complex serverless application. Learners will learn advanced serverless features such as building REST APIs using serverless microservices and stream and event processing. Learners will implement serverless security best practices and then deploy a serverless application.

PROJECT #4

Develop and Deploy with AWS Lambda

In this project learners will develop a serverless application for uploading, listing, and filtering images. They will begin with building serverless REST APIs using API Gateway and AWS Lambda. Then they will use other AWS services (DynamoDB, S3, OpenSearch) to store data and secure the application with authentication. Finally, learners will deploy their app to AWS using a Serverless framework.



Supporting Lesson Content

- Identify the main components of a serverless application
 - Implement a simple application using AWS Lambda
 - Build, package and deploy serverless applications using Serverless Framework
 - Implement HTTP Monitoring by testing code and utilizing AWS Cloudwatch
-
- Build a simple REST API using AWS API Gateway
 - Store and connect to data in DynamoDB
 - Create AWS application resources utilizing AWS CloudFormation
 - Write configuration files leveraging the Serverless Framework
-
- Execute custom logic using lambda functions when triggering events occur
 - Enable a DynamoDB stream
 - Apply distributed tracing to track microservices transactions
 - Implement full-text search using AWS OpenSearch
-
- Implement authentication in a serverless application with Auth0
 - Store secrets for a serverless application using AWS Secrets Manager

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 Instructors



Michał Kapiczyński.

SENIOR SOFTWARE ENGINEER

Michał Kapiczyński is a Senior Software Engineer and an AWS Certified Solutions Architect. His primary expertise is in backend and cloud technologies. He works in consulting helping customers build great cloud solutions and he is part of the Toptal network. He has an Engineering Degree from Aarhus University.



Gabriel Ruttnner

INSTRUCTOR

Gabe is the CTO at Ursa & Tech Advisor for Start-Ups. Gabe has expertise in building cloud-based machine learning and natural language processing services at early stage tech companies. He holds technical degrees from Cornell University and Stony Brook University.

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

**Ivan Mushketyk****PRINCIPAL SOFTWARE ENGINEER**

Ivan Mushketyk is a seasoned Software Engineer with 10 years of experience including companies like AWS and Stripe. He specializes in cloud computing, distributed systems, and data processing. He is an experienced instructor and has created multiple online courses. Ivan is also an avid Open Source contributor.



Learn More at

WWW.UDACITY.COM