

# C S 55J: AWS CERTIFIED SOLUTIONS ARCHITECT ASSOCIATE PREPARATION

## Foothill College Course Outline of Record

| Heading                            | Value   |
|------------------------------------|---|
| <b>Effective Term:</b>             | Summer 2021   |
| <b>Units:</b>                      | 4.5   |
| <b>Hours:</b>                      | 4 lecture, 2 laboratory per week (72 total per quarter)                   |
| <b>Advisory:</b>                   | C S 30A, 55A, 55B, 55C, 55D, 55G, or equivalent knowledge and experience. |
| <b>Degree &amp; Credit Status:</b> | Degree-Applicable Credit Course   |
| <b>Foothill GE:</b>                | Non-GE  |
| <b>Transferable:</b>               | CSU   |
| <b>Grade Type:</b>                 | Letter Grade (Request for Pass/No Pass)                                   |
| <b>Repeatability:</b>              | Not Repeatable  |

## Student Learning Outcomes

- The student will be able to identify the value and benefits of Cloud computing and AWS from the architectural perspective
- The student will demonstrate the use of AWS services to make their cloud infrastructure scalable

## Description

A high level course covering fundamentals of building IT infrastructure on Amazon Web Services (AWS). Designed to teach solutions architects how to optimize the use of the AWS Cloud by understanding AWS services and how these services fit into Cloud-based solutions. Because architectural solutions can differ depending on industry, type of applications, and size of business, this course emphasizes best practices for the AWS Cloud, and it recommends various design patterns to help you think through the process of architecting optimal IT solutions on AWS. It also presents case studies throughout the course, which showcase how some AWS customers have designed their infrastructures, and the strategies and services that they implemented. This course prepares students to pursue becoming an AWS Certified Solutions Architect Associate using official AWS Academy Cloud Foundations material.

## Course Objectives

The student will be able to:

1. Identify the value and benefits of Cloud computing and AWS from the architectural perspective
2. Explain the financial advantages of the AWS Cloud
3. Navigate the AWS Management Console
4. Differentiate between AWS storage options
5. Describe and use AWS database options
6. Identify deployment and management options
7. Make decisions based on recommended architectural principles and best practices

8. Create a cloud migration roadmap and plan
9. Use AWS to make infrastructure scalable
10. Create a business continuity plan and achieve high availability
11. Create event-driven scaling
12. Automate the deployment of Cloud resources
13. Optimize data storage for static content
14. Identify best practices prescribed in the Well-Architected Framework
15. Identify and solve common configuration and design issues
16. Differentiate between architectures for data processing and analytics, mobile back-ends, and video transcoding

## Course Content

1. Environment design
  - a. Regions choice
  - b. Availability zones
  - c. Virtual Private Cloud (VPC)
  - d. Dividing VPCs and default subnets
  - e. Controlling VPC traffic
  - f. Connecting multiple VPCs
  - g. Integrating on-premises components
2. Designing for high availability
  - a. High availability
    - i. Load balancing and fault tolerance
    - ii. High availability across regions
    - iii. Connections outside of AWS
  - b. Scaling
    - i. Best practices
    - ii. Determining need for scaling
    - iii. Automatic scaling
    - iv. Scaling data stores
    - v. AWS Lambda and event-driven scaling
3. Infrastructure automation
  - a. Manual environment configuration
  - b. Infrastructure as code on AWS
  - c. Grouping resources in a template
  - d. Resources not supported by AWS CloudFormation
4. Infrastructure decoupling
  - a. Loose coupling
  - b. Loose coupling strategies
  - c. Communications easily and reliably between components
  - d. Communicating with loose coupling and Amazon DynamoDB
  - e. Amazon API Gateway
  - f. Serverless architectures
  - g. Decoupling examples
5. Designing web-scale media
  - a. Storing web-accessible content with Amazon S3
  - b. Caching with Amazon CloudFront
  - c. Managing NoSQL databases
  - d. Storing relational data in Amazon RDS
6. Well-Architected Framework
  - a. Pillars of the Well-Architected Framework
  - b. Well-Architected design principles
  - c. Pillars

- i. Operational excellence
  - ii. Security
  - iii. Reliability
  - iv. Performance efficiency
  - v. Cost optimization
7. Troubleshooting
  8. Support options
  9. Design patterns and sample architectures
    - a. High availability patterns
    - b. Stream processing
    - c. Sensor network data ingestion and processing
    - d. Back end processing
    - e. Transcoding and serving video files

## Lab Content

1. Making your environment highly available
2. Using notifications to trigger AWS Lambda
3. Automating infrastructure deployment with AWS CloudFormation
4. Implementing a serverless architecture with AWS Managed Services
5. Introducing Amazon CloudFront
6. Multi-region failover with Amazon Route 53

## Special Facilities and/or Equipment

1. Access to a computer with a web browser compatible with the Foothill learning management system.
2. A learning management system with an assignment posting component (through which all lab assignments are to be submitted) and a forum component (where students can discuss course material and receive help from the instructor). This applies to all sections, including on campus (i.e., face-to-face) offerings.
3. The college will provide a fully functional and maintained course management system through which the instructor and students can interact.
4. Students must have email accounts and ongoing access to computers with internet capabilities.

## Method(s) of Evaluation

Subject assessments and quizzes

Laboratory assignments which include detailed instructions, sample runs and documentation

Final examination

## Method(s) of Instruction

Lectures which include motivation for the architecture of the specific topics being discussed

In-person or online labs (for all sections, including those meeting face-to-face/on campus), consisting of:

1. An assignment webpage located on a college-hosted course management system or other department-approved internet environment. Here, the students will review the specification of each assignment and submit their completed lab work
2. A discussion webpage located on a college-hosted course management system or other department-approved internet environment. Here, students can request assistance from the instructor and interact publicly with other class members

Detailed review of laboratory assignments which includes model solutions and specific comments on the student submissions

In-person or online discussion which engages students and instructor in an ongoing dialog pertaining to all aspects of designing, implementing and analyzing programs

When course is taught fully online:

1. Instructor-authored lecture materials, handouts, syllabus, assignments, tests, and other relevant course material will be delivered through a college-hosted course management system or other department-approved internet environment
2. Additional instructional guidelines for this course are listed in the attached addendum of CS department online practices

## Representative Text(s) and Other Materials

Bahga, Arshdeep, and Vijay Madiseti. [Cloud Computing Solutions Architect: A Hands-On Approach, A Competency-based Textbook for Universities and a Guide for AWS Cloud Certification and Beyond](#). 2019.

## Types and/or Examples of Required Reading, Writing, and Outside of Class Assignments

1. Reading:
  - a. Textbook assigned reading averaging 30 pages per week.
  - b. Reading the supplied handouts and modules averaging 10 pages per week.
  - c. Reading online resources as directed by instructor though links pertinent to programming.
  - d. Reading library and reference material directed by instructor through course handouts.
2. Writing:
  - a. Writing technical prose documentation that supports and describes the programs that are submitted for grades.

## Discipline(s)

Computer Science