

C S 55C: COMPUTE ENGINES IN AMAZON WEB SERVICES

Foothill College Course Outline of Record

Heading	Value
Units:	4.5
Hours:	4 lecture, 2 laboratory per week (72 total per quarter)
Advisory:	C S 55A.
Degree & Credit Status:	Degree-Applicable Credit Course
Foothill GE:	Non-GE
Transferable:	CSU
Grade Type:	Letter Grade (Request for Pass/No Pass)
Repeatability:	Not Repeatable

Student Learning Outcomes

- A successful student will be able to explain and apply principles of AWS Docker Containers via Elastic Container Service (ECS).
- A successful student will be able to manage scaling of web servers using AWS Elastic Load Balancer.
- A successful student will be able to demonstrate the principles of Serverless Architectures and Microservices using AWS Lambda Services

Description

In this course, students explore how cloud computing systems are built using a common set of core technologies, algorithms, and design principles centered around distributed systems. Students will use the Amazon Web Services (AWS) Management Console to provision, load-balance and scale their applications using the Elastic Compute Cloud (EC2) and the AWS Elastic Beanstalk. The course discusses, from a developer perspective, the most important reasons for using AWS and examines the underlying design principles of scalable cloud applications.

Course Objectives

The student will be able to:

- Obtain and manage an Amazon (AWS) account and an Amazon Educate account
- Describe basic cloud computing concepts and AWS services
- Apply basic AWS storage services
- Understand basics of AWS EC2 compute instances
- Explain and apply principles of AWS Docker Containers via Elastic Container Service (ECS)
- Demonstrate the principles of Serverless Architectures and Microservices using AWS Lambda Services
- Explain principles of applications as environments via AWS Elastic Beanstalk
- Understand and manage scaling of web servers using AWS Elastic Load Balancer
- Demonstrate principles of provisioning and managing infrastructure using AWS Cloud Formation

Course Content

- AWS access

- AWS account acquisition
- AWS command line interface
- AWS Educate account acquisition
- AWS cloud computing concepts
 - Business drivers
 - Cloud computing models
 - Security and compliance
- AWS cloud platform overview
- AWS storage services
 - AWS Simple Storage Service (S3) object storage
 - Creation and management of S3 Buckets via CLI
 - AWS Glacier archive storage
 - AWS Elastic File System (EFS) network file storage
 - Creation of EFS instances via AWS console
 - AWS Elastic Block Store (EBS)
- AWS Snowball data transport service
- AWS CloudFront content distribution service
- AWS EC2 instances
 - Amazon Machine Instance (AMI) instance types
 - Machine sizing
 - System image types
 - Secure login via key pairs
 - Storage volumes
 - Availability zones
 - IP addressing
 - Virtual network
 - Pricing considerations
 - Security groups
- AWS Docker Containers
 - Concepts
 - Container benefits
 - ECS Container Service
 - Creation and deployment of Docker images with ECS
- Serverless Architectures and Microservices
 - Concepts and benefits (Functions as a Service)
 - AWS Lambda programming
 - Serverless best practices
 - Sample programming with Lambda functions employing DynamoDB
- Application environments using AWS Elastic Beanstalk
 - Create an example Node.js web app using Elastic Beanstalk
 - Manage the application
- Scaling of web servers using AWS Elastic Load Balancing
 - Creation of scaled and load balanced web application via the web interface
 - Creation of scaled and load balanced web application via the CLI
- Basics of AWS Cloud Formation
 - Provisioning of infrastructure via JSON templates

Lab Content

- Create AWS IAM Accounts (Root, Student, Professor)
- Create and manage an AWS S3 Bucket via CLI
- Create Elastic File System (EFS) via the AWS Console
- Setup Linux LAMP EC2 Server with the required security settings
- Setup Linux Server EC2 Instance for use as a template for the other servers in the course
- Create and configure a Docker Image and deploy with ECS (Module 4)
- Create a Lambda "Hello World API Gateway Trigger"
- Create a Lambda function to provide a DynamoDB Read Engine
- Use Elastic Beanstalk to create, view and deploy a simple Node.js web app

J. Configure and deploy a Classic Elastic Load Balancer and Auto Scaling web servers via the web interface

K. Classic Elastic Load Balancer and Auto Scaling web servers via the CLI and then setup a Scaled and Load-Balanced Application via Command Line Interface

Special Facilities and/or Equipment

- A. Access to a computer laboratory with web browsers.
- B. Website or course 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.
- C. When taught via Foothill Global Access on the internet, the college will provide a fully functional and maintained course management system through which the instructor and students can interact.
- D. When taught via Foothill Global Access on the internet, students must have currently existing email accounts and ongoing access to computers with internet capabilities.

Method(s) of Evaluation

- A. Tests and quizzes
- B. Written laboratory assignments, which include detailed instructions, sample runs and documentation
- C. Final examination

Method(s) of Instruction

- A. Lectures which include motivation for the architecture of the specific topics being discussed.
- B. 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.
- C. Detailed review of laboratory assignments which includes model solutions and specific comments on the student submissions.
- D. In person or online discussion which engages students and instructor in an ongoing dialog pertaining to all aspects of designing, implementing and analyzing programs.
- E. 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 C S department online practices.

Representative Text(s) and Other Materials

Ryan, M., and F. Lucifredi. [AWS System Administration](#). O'Reilly Publishers, 2018. ISBN-13: 978-1449342579.

Sarkar, Aurobindo, and Amit Shah. [Learning AWS: Design, Build, and Deploy Responsive Applications using AWS Cloud Components](#), 2nd ed. Packt Publishing, 2018. ISBN-13: 978-1787281066.

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

A. Reading:

1. Textbook assigned reading averaging 30 pages per week.
2. Reading the supplied handouts and modules averaging 10 pages per week.
3. Reading online resources as directed by instructor though links pertinent to programming.
4. Reading library and reference material directed by instructor through course handouts.

B. Writing:

1. Writing technical prose documentation that supports and describes the programs that are submitted for grades.

Discipline(s)

Computer Science