

ITIS 180C: COMPUTE ENGINES IN AMAZON WEB SERVICES

Citrus College Course Outline of Record

Heading	Value
Effective Term:	Fall 2022
Credits:	3
Total Contact Hours:	54
Lecture Hours :	54
Lab Hours:	0
Hours Arranged:	0
Outside of Class Hours:	108
Prerequisite:	ITIS 180A; CS 140 or CS 112.
Transferable to CSU:	No
Transferable to UC:	No
Grading Method:	Standard Letter

Catalog Course 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. 54 lecture hours.

Course Objectives

- Describe important design consideration for scalable cloud applications
- Describe the architectural approach used by AWS
- Navigate the AWS Management Console
- Describe the architectural approach used by AWS' Elastic Beanstalk
- Deploy and manage Elastic Beanstalk applications
- Scale and Load-Balance cloud application using AWS tools
- Deploy EC2 Servers and work with various Amazon Machine Images

Major Course Content

1. Introduction to AWS and the Management Console, Regions and Availability Zones
2. Design Principles for Cloud Applications and Best Practices
3. Architectural Overview of AWS and the Elastic Beanstalk Approach
4. Working With The Elastic Beanstalk
5. Configuring Auto-Scaling and Load Balancing
6. Working With A Git Repository and the EB CLI
7. Deploying A Server With The EC2 Dashboard
8. Configuring An Amazon Machine Image (AMI)
9. Monitoring and Logging with Cloud Watch

Examples of Outside Assignments

Design and write code to implement a tic-tac-toe game using loops that support the AWSConsole and Deployment API. 1. create a new Amazon EC2 server instance from an existing server template 2. create a new security group to restrict access to the server's resource 3. launch the instance 4. access the instance's command-line interface directly, using a key pair for authentication 5. associate an elastic IP address with this EC2 instance 6. deploy code into this EC2 instance that implements this game 7. access the game from a javascript web page that connects via the EC2 instance and its elastic IP address

Instruction Type(s)

Lecture, Online Education Lecture