Course Acronym:	CIS
Course Number:	154
Descriptive Title:	Compute Engines with AWS
Division:	Business
Department:	Computer Information Systems
Course Disciplines:	Computer Information Systems
Catalog Description:	This course introduces compute engine technologies with Amazon Web Services (AWS). Students will explore the following compute models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Function as a Service (FaaS), and microservices. In the course, students will use AWS to set up, configure and manage compute services to address load balancing and scaling requirements in the cloud. Students will also write code to create deployment scripts for the AWS infrastructure.
Prerequisite:	CIS 150 with a minimum grade of C or equivalent experience
Co-requisite:	
Recommended Preparation:	Computer Information Systems 152 (or concurrent enrollment) OR Computer Information Systems 133 with a minimum grade of C or equivalent experience, OR Computer Information Systems 134 with a minimum grade of C or equivalent experience, OR Computer Science 3 with a minimum grade of C or equivalent experience
Enrollment Limitation:	
Hours Lecture (per week):	2
Hours Laboratory (per week):	3
Outside Study Hours:	4
Total Course Hours:	90
Course Units:	3
Grading Method:	Letter Grade only
Credit Status:	Credit, degree applicable
Transfer CSU:	Yes
Effective Date:	Proposed
Transfer UC:	No
Effective Date:	
General Education: ECC	
Term:	
Other:	

CSU GE:	
Term:	
Other:	
IGETC:	
Term:	
Other:	
Student Learning Outcomes:	**SLO #1 AWS Global Infrastructure** Students will be able to demonstrate an understanding of the AWS Global Infrastructure. **SLO #2 AWS Compute Services** Students will be able to set up, configure, and manage a compute engine with AWS and implement load balancing and scaling. **SLO #3 AWS Serverless Applications** Students will be able to create an application using the AWS serverless compute model.
Course Objectives:	1. Describe the design considerations for scalable cloud applications. 2. Explain the AWS Global Infrastructure. 3. Describe important design considerations in migrating services computing. 4. Understand the concept of service load-balancing. 5. Understand the architectural approach used by AWS. 6. Understand AWS Platform as a Service. 7. Understand and configure serverless web applications.
Major Topics:	**I. Design Considerations of Cloud Applications (3 hours. lecture)** A. Introduction to Amazon Web Services (AWS) Compute Services B. Traditional deployments **II. Understanding the AWS Compute Interface (3 hours. lecture)** A. AWS Management Console **III. Infrastructure as a Service (IaaS) (9 hours, lecture)** A. Infrastructure as a Service (IaaS) B. The AWS Architectural approach C. AWS Global Infrastructure D. AWS Infrastructure as Code with AWS CloudFormation **IV. Platform as a Service (PaaS) (9 hours, lecture)** A. AWS Elastic Beanstalk B. Load Balancing Cloud Applications C. Scalable Cloud Applications **V. Function as a Service (FaaS) (9 hours, lecture)** A. Serverless Application Architecture B. AWS Lambda and microservices C. Serverless web applications **VI. Other AWS Compute Considerations (3 hours, lecture)** A. Compute Resource Security

B. AWS Compute Best Practices
C. Migration Considerations

VII. Design Considerations of Cloud Applications (3 hours, lab)
A. AWS Compute Services
B. Traditional deployments
C. The AWS Management Console

VIII. Infrastructure as a Service (IaaS) (15 hours, lab)
A. Infrastructure as a Service (IaaS)
B. The AWS Architectural approach
C. AWS Global Infrastructure
D. AWS Infrastructure as Code with AWS CloudFormation

IX. Platform as a Service (PaaS) (15 hours, lab)
A. AWS Elastic Beanstalk
B. Load Balancing Cloud Applications
C. Scalable Cloud Applications

X. Function as a Service (FaaS) (15 hours, lab)
A. Serverless Application Architecture
B. AWS Lambda and microservices
C. Serverless web applications

XI. Other AWS Compute Considerations (6 hours, lab)
A. Compute Resource Security
B. AWS Compute Best Practices
C. Migration Considerations

Total Lecture Hours:	36
Total Laboratory Hours:	54
Total Hours:	90
Primary Method of Evaluation:	2) Problem solving demonstrations (computational or non-computational)
Typical Assignment Using Primary Method of Evaluation:	Compare and contrast a serverless architecture application with a traditional server-based application. What are the advantages and disadvantages of a serverless approach to deploying applications? Write a one- to two-page report that explores an application where a server based deployment method is more appropriate. Also include in your report an example of an of application where using a serverless platform is preferred.
Critical Thinking Assignment 1:	Using the AWS Console, deploy a game high score calculator which will involve: a. Creating a new Amazon EC2 server instance from an existing server template b. Creating a new security group to restrict access to the server's resource c. Launching the instance d. Accessing the instance's command-line-interface directly, using a key pair for authentication

	e. Associating an elastic IP address with this EC2 instance
	f. Deploying code into this EC2 instance that implements this high score calculator
	g. Accessing the calculator via the EC2 instance and its elastic IP address
Critical Thinking Assignment 2:	Design and write code to implement a tic-tac-toe game using loops that support the AWS Console and Deployment API. You will:
	a. Create a new Amazon EC2 server instance from an existing server template
	b. Create a new security group to restrict access to the server's resource
	c. Launch the instance
	d. Access the instance's command-line-interface directly, using a key pair for authentication
	e. Associate an elastic IP address with this EC2 instance
	f. Deploy code into this EC2 instance that implements this game
	g. Access the game from a JavaScript web page that connects via the EC2 instance and its elastic IP address
Other Evaluation Methods:	Homework Problems, Laboratory Reports, Objective Exam, Quizzes, Written Homework
Instructional Methods:	Demonstration, Discussion, Group Activities, Lab, Lecture, Multimedia presentations
If other:	
Work Outside of Class:	Answer questions, Problem solving activity, Required reading, Skill practice, Written work (such as essay/composition/report/analysis/research)
If Other:	
Up-To-Date Representative Textbooks:	Ryan, M., <u>AWS System Administration</u>, O'Reilly Publishing, 2015. (Discipline Standard) Murty, J., <u>Programming Amazon Web Services</u>, O'Reilly Publishing, 2014. (Discipline Standard)
Alternative Textbooks:	
Required Supplementary Readings:	
Other Required Materials:	
Requisite:	Prerequisite
Category:	sequential
Requisite course(s): List both prerequisites and corequisites in this box.	CIS 150 with a minimum grade of C

Requisite and Matching skill(s):Bold the requisite skill. List the corresponding course objective under each skill(s).	**This course requires an understanding of computer information systems and cloud computing concepts. Students should be able to create cloud applications in AWS.** CIS 150 - Describe the Cloud Computing Model; Create a cloud application utilizing AWS Computing Services (EC2).
Requisite Skill:	Equivalent experience
Requisite Skill and Matching Skill(s): Bold the requisite skill(s). If applicable	Demonstrate an understanding of the development and use of information systems in business.
Requisite course:	Computer Information Systems 152 (or concurrent enrollment) OR Computer Information Systems 133, OR Computer Information Systems 134, OR Computer Science 3
Requisite and Matching skill(s):Bold the requisite skill. List the corresponding course objective under each skill(s).	**Student needs to understand how to connect and interface with AWS databases from a programming environment.** CIS 152 - Compare and contrast the various AWS database services platform. **This class requires assignments to be completed using basic level programming in the JavaScript, C# or Java programming language.** CIS 133 - Design and code web pages using markup languages, scripting language, and web services. CIS 134 - Code an application using conditional structures, variables, classes, functions, procedures, arrays, and database. CSCI 3 - Develop programming code using control and iteration statements; CSCI 3 - Develop programming code using strings and arrays, both single and multidimensional.
Requisite Skill:	
Requisite Skill and Matching skill(s): Bold the requisite skill. List the corresponding course objective under each skill(s). If applicable	
Enrollment Limitations and Category:	
Enrollment Limitations Impact:	
Course Created by:	Khai Lu
Date:	10/18/2018
Original Board Approval Date:	

Last Reviewed and/or Revised by:	
Date:	
Last Board Approval Date:	12/19/2022