

[Home](#) / [My courses](#) / [Cloud Applications Architecture](#) / [Lab](#) / [Lab Exam 12.01.2022/2](#)

Started on Wednesday, 12 January 2022, 4:48 PM

State Finished

Completed on Wednesday, 12 January 2022, 5:44 PM

Time taken 56 mins 14 secs

Grade Not yet graded

Question **1**

Correct

Mark 0.50 out of 0.50

CloudFront is a(n) ✓ and is composed out of ✓ .

Your answer is correct.

Question **2**

Correct

Mark 0.50 out of 0.50

RDS can

Select one or more:

- ☒ upgrade the RDBMS automatically ✓
- ☒ backup the database automatically ✓
- ☒ encrypt the data ✓

Your answer is correct.

Question **3**

Correct

Mark 0.50 out of 0.50

Software can be automatically installed during the provisioning of the EC2 instance through ✓ which is composed of ✓ .

Your answer is correct.

Question **4**

Correct

Mark 0.50 out of 0.50

The most cost-effective and scalable way to host a static web application on AWS is

- ☒ Simple Storage Service (S3)
- ☐ Elastic Container Service (ECS)
- ☐ Elastic Compute Cloud (EC2)



Your answer is correct.

Question **5**

Correct

Mark 0.50 out of 0.50

Docker is a

- ☐ framework
- ☒ container engine
- ☐ programming language



Your answer is correct.

Question **6**

Correct

Mark 0.50 out of 0.50

An EC2 instance is deployed in a specific AZ

Select one:

- ☒ True
- ☐ False



Your answer is correct.

Question 7

Correct

Mark 0.50 out of 0.50

The AWS service for load balancing supports the following types of load balancers:

- Application load balancer which operates at OSI Layer 7 (HTTP/WebSockets) ✓
- Network load balancer which operates at OSI Layer 4 (TCP/UDP) ✓

Your answer is correct.

Question 8

Correct

Mark 0.50 out of 0.50

Simple Storage Service (S3) helps us

Select one:

- ☒ Store and serve an infinite amount of objects such as images, documents, and videos ✓
- ☐ Host web applications
- ☐ Create bootable virtual disks that can be attached to EC2 instances

Your answer is correct.

Question 9

Correct

Mark 0.50 out of 0.50

Which service models does Amazon Web Services provide?

Select one or more:

- ☒ IaaS ✓
- ☒ PaaS ✓
- ☒ SaaS ✓

Your answer is correct.

Question **10**

Correct

Mark 0.50 out of 0.50

An EC2 instance runs in a specific ✓ based on which ✓ IP addresses are assigned to the instance.

Your answer is correct.

Question **11**

Correct

Mark 0.50 out of 0.50

Objects stored in S3 can be made public by configuring the ✓ ✓ .

Your answer is correct.

Question **12**

Correct

Mark 0.50 out of 0.50

You are designing a three-tier web application and want to restrict access to the database tier (remember PostgreSQL instance) to accept traffic from the application servers only (Spring Java app). However, these application servers are in an Auto Scaling group and may vary in quantity.

How should you configure the database servers to meet the requirements?

- ☐ Configure the database subnet network ACL to deny all inbound non-database traffic from the application-tier subnet.
- ☒ Configure the database security group to allow database traffic from the application server security group. ✓
- ☐ Configure the database security group to allow database traffic from the application server IP addresses.
- ☐ Configure the database subnet network ACL to allow inbound database traffic from the application-tier subnet.

Your answer is correct.

Question **13**

Correct

Mark 0.50 out of 0.50

Your application tier is running in an Auto Scaling Group and you need to change the instance type. In which of the following area can this be achieved?

- ☐ Auto scaling policy configuration
- ☒ Auto scaling launch template/configuration
- ☐ Auto scaling AMI
- ☐ Auto scaling tags configuration



Your answer is correct.

Question **14**

Correct

Mark 0.50 out of 0.50

The connection times out while trying to access an EC2 instance via SSH on the standard port. What is the most likely issue?

- ☐ The instance type is misconfigured
- ☐ The instance is not reachable via the internet
- ☒ The security group does not explicitly allow traffic on port 22



Your answer is correct.

Question 15

Complete

Marked out of 3.00

Your organization is running a business-critical learning management system (LMS) on its own infrastructure on-premises. Since the platform has been growing exponentially, the stakeholders decided to migrate it to AWS. You are part of the development team and you are responsible for choosing the right services and designing the solution.

The system enables teachers and experts to market, sell, and present their courses to people willing to learn from all around the world. Each course might contain written documents (mostly PDFs and PowerPoint presentations), videos, and images/diagrams. Moreover, students can be notified via email, each course goes through a review process before being published, and teachers have access to the API of the system enabling them to publish the courses on their own websites.

The LMS is composed out of several backend modules written in Java and Node.js which store data in MySQL databases, while the frontend is built with React. Your team has been struggling lately to run the system locally due to its distributed nature and increasing number of dependencies. Another important consideration is that the system transcodes the videos before the course is published in order to optimize bandwidth and playback.

Since this is the core product of your organization, the main focus of the design must be the availability of the system and reducing operational/maintenance overhead. Of course, cost must be optimized whenever possible.

Describe your solution below. Mention which AWS services you would choose, how they communicate to each other, and, equally important, why you have chosen a particular service. Be as explicit as possible.

Optionally, you can upload images to illustrate your design (feel free to use any drawing or diagram tool you wish; images don't have to be pretty).

Since the stakeholders want the content of the application to be available to all the users across the world, the architecture should make use of a CDN that can serve files as fast as possible around the globe. AWS's CDN is CloudFront. We can use CloudFront to distribute any HTTP accessible data. For this, the architecture should include hosting the application on the cloud. AWS EC2 Service can be used to create virtual machines where the code could live, saving the client to bother about the hardware. Moreover, to increase performance and security, we could use an S3 bucket for the frontend (which can be also accessed through CloudFront by users around the globe), and separate the backend and the database into different storage blocks. The database could be placed inside a private subnet, and through a security group, it should be accessible only by the backend. The backend could be placed in a public subnet inside a VPC and inside several EC2 instances, to increase availability and scalability. An Application Load Balancer could be configured to distribute the load across the EC2 instances, which can be managed through an Auto Scaling Group. This way one can control the cost and be scalable at the same time. For example, maybe in the exam period, more users would access the coursing platform compared to the summer holiday. To address this, ASG could increase or decrease the number of running EC2 instances. Both the ALB and the ASG together with its instances should be wrapped in security groups allowing specific traffic of specific types, from and to specific targets. For example, the EC2 instances should only accept traffic from the Load Balancer through its Target Group.

To make it all easier to manage, we could use AWS services that can run and manage the database for us (RDS) or run our application together

with all its dependencies (Docker). This means that we no longer have to worry about keeping the database online, applying updates and backups. Another convenience is that RDS supports the database used by the given application to store its data (MySQL). To manage the backend, we could use Amazon's proprietary container orchestration service, ECS. By creating an ECS service, we will ensure that a predefined number of container instances is available at all times. The service also exposes the containers through a load balancer. For sending emails to the users the architecture could include an SES, which is cost-effective, flexible, and scalable, allowing the application to send configurable mails. To deal with video transcoding, the architecture could use the following solution: use Amazon Glacier to store the master videos, Amazon Elastic Transcoder to transcode masters to renditions for distribution stored in an Amazon S3 bucket and then stream these renditions at scale, to all users, over the Internet using CloudFront. We could also use CloudWatch to get real-time monitoring and metrics. To deal with user management we could use AWS Cognito service together with a user pool to give access to teachers and experts to upload courses. Then, by using API Gateway, we can add a new endpoint for /course/review. This is intended to be called by teachers after they

deliver new courses. We will send/buffer all requests to an SQS queue from where a lambda

function will read and process them, ultimately calling the SES service to inform the reviewers that a recently uploaded course needs a review before it is published. Same for the reviewers, they could access a new endpoint course/publish, and through the same process, an SES will send emails to users that a new course has been published.

All this could be done using a CloudFormation stack, which makes the transition from a "legacy" application to a fully cloud-native architecture a lot easier.

[Untitled.png](#)[video.png](#)

◀ [Resource Cleanup](#)

Jump to...