Evaluación del módulo #4

Consigna del proyecto



Evaluación del módulo

Proyecto: Infraestructura Viva

Situación inicial 📍



Unidad solicitante: Equipo de Innovación Tecnológica de la empresa "Soluciones Digitales ACME"

El área de Innovación Tecnológica ha identificado la necesidad de modernizar la infraestructura para responder a crecientes requerimientos de procesamiento, almacenamiento y disponibilidad de datos. Actualmente, los departamentos de ventas, soporte y finanzas utilizan un entorno on-premise que genera altos costos de mantenimiento y limita la rápida implementación de nuevas funcionalidades.

Las problemáticas a resolver son: 👇



- 1. Disponer de un entorno de cómputo escalable y seguro para desplegar aplicaciones.
- 2. Integrar de forma unificada bases de datos relacionales y NoSQL sin depender de hardware físico.
- 3. Implementar un sistema de almacenamiento confiable, diferenciando datos de uso frecuente y de archivado.
- 4. Configurar una red virtual que permita aislar servicios internos y exponer aplicaciones públicas de forma segura.
- 5. Establecer un plan de monitoreo y notificaciones que detecte y corrija incidentes de forma proactiva.
- 6. Diseñar una solución que se pueda implementar utilizando los contenidos y conceptos vistos en los manuales.

Además, se considerará el uso de recursos de AWS Academy cuando se requiera mayor profundidad en aspectos cloud, aprovechando sus laboratorios y materiales gratuitos para complementar la propuesta.



Nuestro objetivo 📋

Desarrollar la propuesta "Infraestructura Viva", que contemple la migración del entorno on-premise a la nube, utilizando herramientas gratuitas y recursos de AWS Academy. La solución debe estar basada en los conceptos vistos en los manuales, abarcando:

- **Cómputo:** Despliegue de aplicaciones mediante máquinas virtuales, contenedores o funciones serverless (free tier y AWS Academy).
- Bases de Datos: Integración de un motor relacional gestionado y una solución NoSQL, con pruebas realizadas en SQLiteOnline para la parte SQL.
- Almacenamiento: Uso de almacenamiento de objetos para archivos estáticos y opciones de respaldo/archivo (Glacier o equivalente).
- **Red:** Configuración de una red virtual privada (VPC) con subredes públicas y privadas, reglas de seguridad y balanceo de carga.
- Monitoreo y Notificaciones: Implementación de un sistema de monitoreo (CloudWatch o similar) y notificaciones (SNS/SQS) para la detección y respuesta a incidentes.

Requerimientos 🤝

1. Cómputo

- Desplegar aplicaciones en entornos de máquinas virtuales, contenedores o funciones serverless (aprovechando el AWS Free Tier y los recursos de AWS Academy)
- Utilizar planes gratuitos o de prueba para minimizar costos.

2. Bases de Datos

- Utilizar un servicio de base de datos relacional (tipo RDS o similar) para datos transaccionales.
- Incluir una solución NoSQL (por ejemplo, DynamoDB o equivalente) para datos semiestructurados.



 Realizar pruebas y prototipos de consultas SQL utilizando SQLiteOnline.

3. Almacenamiento

- o Implementar almacenamiento de objetos (ej. S3 o similar) para contenido estático y archivos multimedia.
- Proponer almacenamiento de largo plazo (como Glacier) para respaldos y datos históricos, diferenciando clases de almacenamiento según el uso.

4. Red en la nube

- Configurar una red virtual (VPC o equivalente) con subredes segmentadas para aislar servicios internos y exponer aplicaciones críticas mediante un balanceador de carga.
- Definir reglas de seguridad (firewalls, grupos de seguridad) y, si es necesario, integrar soluciones VPN para conectar el entorno on-premise con la nube.

5. Monitoreo y Notificaciones

- Implementar un servicio de monitoreo (como CloudWatch o equivalente) para recopilar métricas de cómputo, bases de datos y red.
- Configurar alarmas (por ejemplo, umbrales de CPU o errores en logs) y notificaciones automáticas a través de canales como email, SMS o integraciones con plataformas de chat.
- Correlacionar eventos para desencadenar acciones automáticas de respuesta ante incidentes.

6. Notificaciones y mensajería

- Establecer un canal de mensajería (SNS, SQS o similar) que facilite la comunicación entre microservicios y el envío de alertas a los equipos de soporte.
- o Garantizar la alta disponibilidad de la mensajería y definir protocolos de reintentos.



7. Herramientas gratuitas y de apoyo

- Visual Studio Code: Para la edición de código, configuración de laC y scripts de automatización.
- SQLiteOnline: Para ejecutar y probar consultas SQL sin necesidad de instalaciones locales.
- AWS Academy: Como recurso adicional para aprender y aplicar conceptos cloud mediante laboratorios y materiales gratuitos, complementando lo visto en los manuales.

Paso a paso 👣

Este proyecto refiere exclusivamente al **módulo 4**: Fundamentos de tecnología Cloud, y se compone de **8 etapas (lecciones)**, las cuales podrás avanzar de forma progresiva y escalonada con la ayuda de los manuales teóricos y los contenidos desarrollados en las clases en vivo.

Considera invertir **tiempo asincrónico** para el desarrollo de cada etapa a modo de poder finalizar el módulo y realizar la entrega formal de tu propuesta. Cualquier consulta que surja compártela en los espacios sincrónicos para resolver las dudas en equipo.

A continuación encontrarás las consignas y tareas a desarrollar:

- Lección 1 Almacenamiento en Cloud
 - **Objetivo:** Implementar estrategias de almacenamiento en la nube diferenciando entre datos de uso frecuente y archivado.

Tareas a desarrollar:

- Configurar un bucket en Amazon S3 para almacenar archivos estáticos.
- 2. Crear una política de ciclo de vida para mover archivos antiguos a almacenamiento de archivo (ej. S3 Glacier).
- 3. Definir una estrategia de respaldo automático (manual o con scripts).
- **4.** Justificar el tipo de almacenamiento elegido (por uso, costo y disponibilidad).



→ Métricas mínimas y máximas:

- → 1 bucket estándar + 1 bucket archivado (mínimo).
- → Hasta 3 buckets con políticas diferentes (máximo).
- → 1 política de backup implementada.

Lección 2 – Bases de Datos Relacionales

Objetivo: Implementar una base de datos relacional gestionada y realizar pruebas de SQL.

Tareas a desarrollar:

- 1. Leer el Manual #2: Servicios de bases de datos relacionales.
- Desplegar una instancia de Amazon RDS o similar (MySQL, PostgreSQL).
- 3. Configurar alta disponibilidad y backups automáticos.
- 4. Realizar pruebas de consultas básicas en **SQLiteOnline**.
- 5. Documentar configuración, costos estimados y escalabilidad.

→ Métricas:

- → 1 instancia RDS con backups configurados.
- → 3-5 consultas SQL probadas en SQLiteOnline.
- → 1 documentación de configuración RDS.

Lección 3 – Bases de Datos NoSQL

Objetivo: Desplegar e integrar una base NoSQL para manejar datos semiestructurados.

Tareas a desarrollar:

- 1. Leer el Manual #3: Servicios de bases de datos NoSQL.
- 2. Crear una tabla en Amazon DynamoDB o equivalente.
- 3. Configurar índices secundarios para consultas rápidas.
- 4. Integrar DynamoDB con una aplicación de ejemplo.
- 5. Definir una estrategia de respaldo para esta base.

→ Métricas:

- 1 tabla NoSQL creada.
- → Hasta 3 índices configurados.
- → 1 integración funcional con una aplicación.



- **Lección 4** Servicios de Cómputo
 - **Objetivo:** Desplegar una aplicación utilizando recursos de cómputo escalables.

Tareas a desarrollar:

- 1. Leer el Manual #4: Servicios de cómputo.
- Desplegar una app en EC2, ECS o Lambda (mínimo una opción).
- 3. Realizar pruebas de escalabilidad o ejecución bajo carga.
- 4. Utilizar el Free Tier de AWS o AWS Academy.

→ Métricas:

- → 1 instancia o función serverless desplegada (mínimo).
- → Hasta 3 servicios de cómputo utilizados (máximo).
- → 1 prueba de escalabilidad documentada.
- Lección 5 Servicios de Red en la Nube
 - **Objetivo:** Diseñar una red segura y eficiente para segmentar servicios internos y externos.

Tareas a desarrollar:

- 1. Leer el Manual #5: Servicios de red en la nube.
- Configurar una VPC con al menos 1 subred pública y 1 privada.
- 3. Definir grupos de seguridad, NACLs y, si aplica, una VPN.
- 4. Implementar un balanceador de carga para distribuir tráfico web.

→ Métricas:

- → 1 VPC con 2 subredes (mínimo).
- → Hasta 4 reglas de seguridad configuradas.
- → 1 balanceador de carga activo.
- Lección 6 Notificación y Mensajería
 - **Objetivo:** Configurar servicios de mensajería y alertas para comunicación entre servicios y atención a incidentes



📍 Tareas a desarrollar:

- 1. Leer el Manual #6: Servicios de notificación y mensajería.
- 2. Configurar un tema en Amazon SNS y una cola SQS.
- 3. Integrar SNS con un servicio de correo o mensaje.
- 4. Implementar un patrón de reintentos y alta disponibilidad.

→ Métricas:

- → 1 SNS + 1 SQS funcionales (mínimo).
- → Hasta 3 suscriptores o integraciones externas.
- → 1 política de reintento aplicada.
- Lección 7 Servicios simples de alojamiento web y contenidos
 - **Objetivo:** Publicar contenido estático y distribuirlo globalmente con buena performance.

Tareas a desarrollar:

- Leer el Manual #7: Servicios simples de alojamiento web y contenidos.
- 2. Desplegar un sitio web estático en S3.
- 3. Configurar distribución con Amazon CloudFront (CDN).
- 4. Añadir certificados SSL y dominio personalizado (opcional).

→ Métricas:

- → 1 sitio web público en S3.
- → 1 distribución CloudFront implementada.
- → Hasta 2 configuraciones de seguridad (SSL, permisos).
- Lección 8 Monitoreo y correlación de Incidentes
 - **Objetivo:** Monitorizar servicios y configurar respuestas automáticas ante fallos.

📍 Tareas a desarrollar:

- Leer el Manual #8: Servicios de monitoreo y correlación de incidentes.
- 2. Configurar métricas en Amazon CloudWatch (CPU, red, errores).
- 3. Establecer alarmas con umbrales y acciones automáticas.



4. Documentar protocolos de actuación y correlación de eventos.

→ Métricas:

- → 3 métricas clave monitorizadas.
- → 2 alarmas con notificación SNS.
- → 1 acción automática documentada.

¿Qué vamos a validar? 🔍

Diseño arquitectónico:

La coherencia y cobertura de todos los componentes (cómputo, bases de datos, almacenamiento, red, monitoreo y mensajería) en la solución propuesta.

• Cumplimiento de requerimientos:

La propuesta debe incluir todos los puntos solicitados y estar basada en los conceptos y técnicas vistos en los manuales.

• Buenas prácticas:

Uso adecuado de medidas de seguridad, escalabilidad y alta disponibilidad, incluyendo configuraciones de firewall, roles de acceso y estrategias de replicación.

Viabilidad de implementación:

Que la solución sea factible utilizando herramientas gratuitas (free tier, AWS Academy, SQLiteOnline) y se pueda desplegar siguiendo los contenidos aprendidos en los manuales.

• **Documentación y claridad:** Diagramas y descripciones claras que expliquen el flujo y la integración de cada servicio dentro de la arquitectura.

Referencias 🦺

- Manual #1 Almacenamiento en cloud: Estrategias de almacenamiento de objetos, bloques y sistemas de archivos.
- Manual #2 Servicios de bases de datos relacionales: Conceptos sobre bases de datos gestionadas, escalabilidad y costos.



- Manual #3 Servicios de bases de datos NoSQL: Características y aplicaciones de soluciones NoSQL.
- Manual #4 Servicios de cómputo: Opciones de cómputo (máquinas virtuales, contenedores, serverless) y su escalabilidad.
- Manual #5 Servicios de red en la nube: Configuración de VPC, subredes y seguridad en la red.
- Manual #6 Servicios de notificación y mensajería: Patrones de mensajería y estrategias de notificaciones.
- Manual #7 Servicios simples de alojamiento web y contenidos: Opciones de hosting y distribución de contenidos.
- Manual #8 Servicios de monitoreo y correlación de incidentes: Configuración de métricas, alarmas y correlación de eventos.

Entregables 🔽

1. Documento de arquitectura:

- Descripción detallada de la solución "Infraestructura Viva", justificando el uso de cada servicio y recurso (incluyendo AWS Academy).
- Diagrama representativo de la arquitectura.

2. Prototipo o demo:

- Instrucciones para desplegar una instancia de cómputo (VM o función serverless) conectada a una base de datos gestionada.
- Ejemplos de configuración en Visual Studio Code y demostración de consultas en SQLiteOnline.

3. Plan de monitoreo y notificaciones:

 Configuración de alarmas y notificaciones (por ejemplo, umbrales de CPU y errores en logs).

4. Guía de buenas prácticas:

 Consideraciones de seguridad, escalabilidad y administración de red basadas en los manuales.



5. Presentación final o informe resumido:

- Resumen de la propuesta (3-5 páginas o presentación de 10 minutos) que explique la arquitectura, justificaciones y pasos de implementación.
- Integración del código y documentación en un repositorio público (por ejemplo, GitHub).

Portafolio 💼

Se invita a los participantes a incorporar el proyecto "Infraestructura Viva" en su portafolio profesional, mostrando:

- El diagrama y la explicación de la arquitectura.
- La transición de un entorno on-premise a la nube, resaltando la optimización de costos y la seguridad.
- Los aprendizajes y la aplicación de conceptos de los manuales, junto con el uso de herramientas gratuitas como Visual Studio Code, SQLiteOnline y AWS Academy.
- El código fuente y la documentación en un repositorio público.

También es recomendable agregar una breve reflexión personal sobre los desafíos enfrentados y los aprendizajes obtenidos durante el desarrollo del proyecto.

Esta evidencia mostrará tu capacidad para integrar conceptos de arquitectura, gobernanza, calidad y análisis de datos en una solución real.



¡Éxitos!

Nos vemos más adelante

