**📋 Test de conocimientos — Bloque 2: Fundamental Cloud y microservicios**

Duración estimada: **1 hora**  
Instrucciones: Marca la alternativa correcta. Algunas preguntas tienen ejercicios cortos.

**☁ 1. Arquitectura Cloud (resiliencia, escalabilidad y alta disponibilidad)**

**1.1** ¿Qué principio busca que un sistema pueda atender más usuarios o procesar más datos al aumentar los recursos?

* a) Resiliencia
* b) Escalabilidad
* c) Elasticidad
* d) Disponibilidad

**1.2** ¿Cuál de estas prácticas contribuye directamente a repartir la carga de trabajo y evitar puntos únicos de fallo?

* a) Autoescalado vertical de instancias
* b) Crecimiento horizontal de recursos
* c) Uso de una sola VM por región
* d) Balanceo de carga de peticiones

**1.3** En una arquitectura resiliente, ¿qué se recomienda para tolerar fallos de hardware o zona?

* a) Evitar redundancia para simplificar
* b) Uso de zonas de disponibilidad (AZ)
* c) Procesar todo en una única región
* d) Configurar cada recurso en modo single instance

**🐳 2. Servicios en la nube**

**2.1** ¿Cuál de estos servicios de AWS permite ejecutar código sin administrar servidores?

* a) EC2
* b) S3
* c) Lambda
* d) RDS

**2.2** ¿Cuál servicio de GCP se usa para crear máquinas virtuales?

* a) Cloud Functions
* b) Cloud Storage
* c) Compute Engine
* d) Cloud SQL

**2.3** ¿Qué servicio de Azure corresponde a bases de datos relacionales gestionadas?

* a) Blob Storage
* b) Azure Functions
* c) Azure SQL Database
* d) Cosmos DB

**🔧 3. Prácticas breves**

**3.1** Describe brevemente qué es un bucket en Cloud Storage de GCP.  
*Respuesta libre, 1–2 líneas.*

**3.2** Supón que tienes un sitio web estático y quieres alta disponibilidad global. Marca la combinación de pasos más adecuada:

* a) Crear un bucket multirregional en Cloud Storage
* b) Configurar un balanceador de carga HTTP(S) global
* c) Habilitar CDN (Cloud CDN)
* d) Desplegar la web en una sola VM
* A) a y d
* B) a, b y c
* C) b y c
* D) c y d

**🔐 4. Gestión y seguridad**

**4.1** En GCP, ¿qué permite asignar permisos específicos a usuarios o grupos?

* a) Cloud Shell
* b) IAM
* c) VPC
* d) Cloud Functions

**4.2** ¿Qué elemento organiza recursos y facilita la facturación en GCP?

* a) Proyecto
* b) Región
* c) Bucket
* d) Máquina virtual

**🧩 5. Microservicios (conceptos introductorios)**

**5.1** ¿Qué describe mejor el enfoque de microservicios?

* a) Un único servicio central que hace todo
* b) Dividir la aplicación en servicios pequeños y autónomos
* c) Utilizar funciones como servicio (FaaS) para cada tarea
* d) Ejecutar todo en el mismo proceso

**5.2** ¿Qué propone DDD (Domain-Driven Design)?

* a) Priorizar la infraestructura sobre el dominio
* b) Modelar el software basándose en el negocio real
* c) Crear una única base de datos compartida
* d) No considerar reglas de negocio

**5.3** La arquitectura hexagonal recomienda:

* a) Acoplar directamente la lógica de negocio con APIs externas
* b) Separar el núcleo del dominio de la infraestructura y entradas/salidas
* c) Usar solo bases de datos SQL
* d) Evitar cualquier tipo de abstracción

**⚙ 6. Pregunta práctica final (opcional, bonus)**

**Describe brevemente** cómo crearías una VM con alta disponibilidad en GCP.  
*Respuesta libre, 1–2 líneas.*

**🚀 7. Preguntas adicionales**

Profundiza con estas preguntas enfocadas en Docker básico, GCP y microservicios:

**7.1** ¿Qué comando se usa para construir una imagen Docker a partir de un Dockerfile?

* a) docker run
* b) docker build
* c) docker compose
* d) docker init

**7.2** ¿Qué comando ejecuta un contenedor en segundo plano (“detached”)?

* a) docker start
* b) docker exec
* c) docker run -d
* d) docker launch

**7.3** ¿Qué comando muestra los contenedores en ejecución?

* a) docker images
* b) docker ps
* c) docker list
* d) docker top

**7.4** ¿Qué característica del grupo de instancias manejadas de GCP adapta la capacidad según la demanda?

* a) Autoscaling
* b) Instance Templates
* c) Zonal Redundancy
* d) Static IP

**7.5** ¿Cómo se configura la alta disponibilidad en Cloud SQL de GCP?

* a) Habilitando réplicas de lectura
* b) Configurando una instancia en modo single-zone
* c) Activando una instancia con failover en multi-zona
* d) Instalando un proxy local

**7.6** En funciones serverless, ¿qué problema está asociado al “cold start”?

* a) Latencia inicial al invocar una función no calentada
* b) Costo elevado por ejecución
* c) Falta de escalabilidad
* d) Ausencia de logs

**7.7** ¿Qué es un volumen en Docker?

* a) Un tipo de red
* b) Almacenamiento persistente que se monta en un contenedor
* c) Una imagen ligera
* d) Un proceso secundario

**🔍 8. Casos prácticos**

**8.1** En el banco se realizarán ventas de entradas desde el sitio web y se necesita analizar cómo hacer el sistema escalable. ¿Qué recomendarías para manejar picos de tráfico en el frontend y backend?

* a) Colocar un balanceador de carga frente a múltiples instancias de la aplicación
* b) Usar instancias serverless (Cloud Functions o Lambda) para la capa de backend
* c) Escalar verticalmente la base de datos principal únicamente
* d) Almacenar temporalmente las solicitudes en una cola desacoplada (Cloud Pub/Sub o SQS)
* A) a y b
* B) a, b y d
* C) c y d
* D) b y c

**8.2** ¿Qué ventajas trae el uso de contenedores en el desarrollo y despliegue de microservicios?

* a) Portabilidad entre entornos
* b) Aislamiento de dependencias
* c) Movimiento automático de contenedores sin downtime
* d) Facilidad para escalar servicios individuales
* A) a y b
* B) a, b y d
* C) c y d
* D) todas

**8.3** Un microservicio de procesamiento de pagos debe garantizar consistencia eventual y tolerancia a fallos. ¿Qué patrón o componente recomendarías añadir?

* a) Cola de mensajes (Pub/Sub o SQS)
* b) Base de datos SQL monolítica
* c) Un único servidor con réplica pasiva
* d) Balanceador de carga en capa 7
* A) a
* B) b y c
* C) c y d
* D) d

**8.4** Para asegurar que los logs de varias instancias de una aplicación se centralicen y analicen, ¿qué solución cloud utilizarías?

* a) Almacenamiento de archivos en cada VM
* b) Servicio de logging centralizado (Cloud Logging, ELK)
* c) Envío de logs por correo
* d) Guardar logs en una base de datos relacional
* A) b
* B) a y c
* C) c y d
* D) todas

**8.5** ¿Cuál es una ventaja de utilizar FaaS (Functions as a Service) en arquitecturas de microservicios?

* a) Cobro por uso basado en invocación y tiempo de ejecución
* b) Mantenimiento automático de la infraestructura subyacente
* c) Control total del sistema operativo por parte del desarrollador
* d) Cero costo en periodos de inactividad
* A) a y b
* B) b y c
* C) a y d
* D) c y d

**8.6** Para un servicio REST en contenedores, ¿qué configuraciones asegurarían alta disponibilidad y actualizaciones sin downtime?

* a) Especificar replicas: >1 en el Deployment
* b) Usar readiness y liveness probes
* c) Realizar rolling updates
* d) Crear un StatefulSet con volumen local
* A) a y b
* B) a, b y c
* C) b y d
* D) todas