

# Curso

# Herramientas de Computación en la Nube

## Guía de estudio 1

Febrero 7 de 2026

## Profesor

Alvaro Mauricio Montenegro Díaz

Universidad de la Sabana

## Aplicaciones confiables, escalables y mantenibles

### 1. Propósito del estudio

Comprender los principios que rigen el diseño de aplicaciones orientadas a datos, enfocándose en la **confiabilidad**, la **escalabilidad** y la **mantenibilidad** como atributos arquitectónicos fundamentales, y en los **trade-offs** inherentes a su implementación.

### 2. Aplicaciones orientadas a datos

#### Contenidos clave

- Diferencia entre aplicaciones orientadas a datos y aplicaciones orientadas al cómputo.
- Factores dominantes: volumen, complejidad y velocidad de cambio de los datos.
- Uso de componentes estándar (bases de datos, cachés, índices, mensajería, batch).

#### Preguntas de evaluación

1. ¿Qué características distinguen a una aplicación orientada a datos de una orientada al cómputo?
2. ¿Por qué el uso de múltiples componentes da lugar a sistemas de datos compuestos?
3. ¿Qué implicaciones arquitectónicas surgen al integrar varias tecnologías de datos?

### 3. Confiabilidad

#### Contenidos clave

- Definición de confiabilidad como funcionamiento correcto bajo condiciones adversas.

- Distinción entre falla (fault) y fallo (failure).
- Principio de tolerancia a fallas.
- Fuentes de fallas: hardware, software y error humano.

### **3.1 Fallas de hardware**

- Naturaleza inevitable de las fallas físicas.
- Uso de redundancia y replicación.

### **3.2 Errores de software**

- Errores sistemáticos y correlacionados.
- Riesgo de fallas en cascada.
- Importancia de pruebas, aislamiento y monitoreo.

### **3.3 Error humano**

- Configuración incorrecta y cambios operativos.
- Rol de la automatización y despliegues graduales.

### **Preguntas de evaluación**

4. Explique la diferencia entre falla y fallo con un ejemplo práctico.
5. ¿Por qué la tolerancia a fallas es preferible a la prevención absoluta?
6. Compare el impacto de las fallas de hardware y de software en sistemas distribuidos.
7. ¿Por qué el error humano es una de las principales fuentes de interrupciones?
8. ¿Qué prácticas reducen el impacto de los errores operativos?

## **4. Escalabilidad**

### **Contenidos clave**

- Escalabilidad como propiedad relativa.
- Importancia de describir la carga antes de escalar.
- Medición de rendimiento mediante distribuciones y percentiles.

- Fenómenos de latencia y colas.

#### **4.1 Descripción de carga**

- Solicitudes por segundo.
- Proporción lectura/escritura.
- Distribuciones extremas.

#### **4.2 Medición de rendimiento**

- Uso de percentiles (p50, p95, p99).
- Relevancia de la *tail latency*.

#### **4.3 Estrategias de crecimiento**

- Escalado vertical.
- Escalado horizontal.
- Elasticidad.

### **Preguntas de evaluación**

9. ¿Por qué la escalabilidad no puede describirse como una propiedad absoluta?
10. ¿Qué información debe incluir una descripción adecuada de la carga?
11. ¿Por qué los promedios son métricas engañosas para el rendimiento?
12. Explique el concepto de *tail latency* y su impacto en la experiencia del usuario.
13. Compare escalado vertical y horizontal, indicando ventajas y desventajas.

### **5. Mantenibilidad**

#### **Contenidos clave**

- Importancia del costo post-despliegue.
- Tres pilares: operabilidad, simplicidad y evolvabilidad.

#### **5.1 Operabilidad**

- Observabilidad.

- Automatización.
- Recuperación rápida.
- Gestión de despliegues.

## **5.2 Simplicidad**

- Complejidad accidental.
- Abstracciones claras.
- Reducción del acoplamiento.

## **5.3 Evolvabilidad**

- Cambio constante de requisitos.
- Capacidad de adaptación sin reescrituras masivas.

### **Preguntas de evaluación**

14. ¿Por qué la mantenibilidad es un atributo arquitectónico crítico?
15. ¿Qué responsabilidades recaen en los equipos de operaciones?
16. Defina complejidad accidental y explique cómo puede reducirse.
17. ¿Qué relación existe entre simplicidad y evolvabilidad?
18. Proporcione un ejemplo de diseño que dificulte la evolución de un sistema.

## **6. Síntesis integradora**

### **Contenidos clave**

- Interdependencia entre confiabilidad, escalabilidad y mantenibilidad.
- Necesidad de decisiones conscientes y fundamentadas.
- Ausencia de soluciones universales.

### **Preguntas de evaluación**

19. ¿Por qué estos tres atributos deben considerarse de forma conjunta?
20. Analice un sistema real e identifique compromisos entre confiabilidad, escalabilidad y mantenibilidad.

## **7. Pregunta integradora avanzada**

21. Diseñe a alto nivel una arquitectura para una aplicación orientada a datos, explicando cómo se abordan explícitamente la confiabilidad, la escalabilidad y la mantenibilidad, y qué trade-offs se aceptan.