



**UNIVERSIDAD LATINA  
DE COSTA RICA**

POWERED BY **Arizona State University®**

**Universidad Latina de Costa Rica**

**Ingeniería en Telemática**

**Avance I**

**Proyecto de Investigación Sistemas Operativos II**

**BIT-28 Sistemas Operativos II**

**Jimena Gómez Calvo**

**Docente: Carlos Méndez Rodríguez**

**2025**

## 1. Introducción

En la actualidad, la computación en la nube ha revolucionado la forma en que las organizaciones gestionan sus sistemas y servicios. Según el Instituto Nacional de Estándares y Tecnología (NIST), la computación en la nube se define como “un modelo para permitir el acceso conveniente, bajo demanda y por medio de la red de Internet, a un grupo compartido de recursos de computación configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que se pueden aprovisionar y liberar rápidamente con un mínimo esfuerzo de gestión o interacción con el proveedor de servicios”. Esta tecnología permite una escalabilidad eficiente y una alta disponibilidad, lo que la hace ideal para entornos donde el rendimiento y la seguridad son fundamentales.

Este informe tiene como objetivo analizar y diseñar una arquitectura de sistemas distribuida con alta disponibilidad, escalabilidad, seguridad y alto rendimiento. Se examinará el impacto de diferentes algoritmos de balanceo de carga en el rendimiento del sistema, evaluando métricas clave como el tiempo de respuesta, distribución del tráfico y tiempo de inactividad. Con base en estos resultados, se propondrán mejoras para optimizar la infraestructura y su desempeño.

## 2. Objetivos del Proyecto

1. **Analizar aspectos de alta disponibilidad, alto rendimiento, escalabilidad y seguridad:** Este objetivo implica llevar a cabo un estudio exhaustivo de cada uno de estos aspectos críticos en el diseño de sistemas distribuidos. La alta disponibilidad se refiere a la capacidad del sistema para permanecer operativo y accesible incluso en caso de fallos. El alto rendimiento se centra en la rapidez y eficiencia con la que el sistema puede procesar solicitudes. La escalabilidad se relaciona con la capacidad del sistema para adaptarse a un aumento en la carga de trabajo, permitiendo agregar recursos sin afectar negativamente el rendimiento. Por último, la seguridad abarca las medidas y protocolos necesarios para proteger los datos y las operaciones del sistema de amenazas externas.

2. **Diseñar una arquitectura robusta con los aspectos anteriores:** Este objetivo se enfoca en la creación de una arquitectura de sistema que integre los principios de alta disponibilidad, rendimiento, escalabilidad y seguridad. La arquitectura debe ser capaz de soportar la carga de trabajo esperada y proporcionar un marco flexible que permita adaptaciones a medida que cambien las necesidades del negocio. Esto puede incluir la implementación de redundancia, balanceadores de carga, sistemas de replicación de datos y políticas de seguridad adecuadas.
3. **Evaluar el impacto de diferentes algoritmos de balanceo de carga en el rendimiento del sistema:** En este objetivo, se busca investigar cómo diferentes técnicas de balanceo de carga afectan el rendimiento general del sistema. Se analizarán algoritmos como Round Robin, Least Connections y Weighted Load Balancing para determinar cuál proporciona los mejores resultados en términos de latencia, uso de recursos y eficiencia en la distribución de solicitudes. Esta evaluación será clave para optimizar la respuesta del sistema ante diversas cargas de trabajo.
4. **Determinar los criterios de evaluación y métricas clave de rendimiento como tiempo de respuesta, distribución del tráfico y tiempo de inactividad:** Este objetivo consiste en establecer un conjunto de métricas y criterios que se utilizarán para evaluar el rendimiento del sistema. El tiempo de respuesta medirá la rapidez con la que el sistema responde a las solicitudes de los usuarios. La distribución del tráfico analizará cómo se distribuyen las solicitudes entre los diferentes servidores y cómo se gestiona la carga. El tiempo de inactividad se referirá al tiempo durante el cual el sistema no está disponible para los usuarios. Estas métricas permitirán realizar un seguimiento efectivo del rendimiento y la disponibilidad del sistema.
5. **Proponer mejoras basadas en los resultados obtenidos:** Finalmente, este objetivo implica utilizar los hallazgos de las evaluaciones y análisis realizados para sugerir mejoras en la arquitectura y funcionamiento del

sistema. Estas mejoras pueden incluir la implementación de nuevos algoritmos de balanceo de carga, ajustes en la configuración de recursos, o la incorporación de tecnologías emergentes que optimicen la eficiencia, seguridad y rendimiento del sistema. El objetivo es garantizar que el sistema evolucione para satisfacer las necesidades cambiantes del entorno operativo y de los usuarios.

### 3. Implementación

#### 3.1 Arquitectura Propuesta

La arquitectura propuesta se basa en una combinación de múltiples servidores interconectados, un balanceador de carga y mecanismos de seguridad avanzados. Este diseño permite garantizar la disponibilidad y escalabilidad del sistema en caso de fallas o aumentos inesperados de la demanda.

- **Balanceador de carga:** Se implementará un sistema de distribución de tráfico utilizando algoritmos como Round Robin, Least Connections y Weighted Load Balancing.
- **Alta disponibilidad:** Se aplicará redundancia de servidores y mecanismos de replicación de datos para minimizar el impacto de fallos.
- **Seguridad:** Se incluirán medidas como cortafuegos, detección de intrusos y mitigación de ataques DDoS.

#### 3.2 Entorno y Métricas Configuradas

Para evaluar el desempeño del sistema, se configurará un entorno de pruebas basado en herramientas de monitoreo. Se medirán los siguientes indicadores clave:

- **Tiempo de respuesta:** Latencia desde que un usuario realiza una solicitud hasta que recibe una respuesta.
- **Distribución del tráfico:** Evaluación de cómo el balanceador de carga distribuye peticiones entre los servidores.

- **Tiempo de inactividad:** Cálculo del porcentaje de disponibilidad del sistema.

#### **4. Resultados y Discusión**

Los resultados obtenidos permitirán comparar la eficiencia de los diferentes algoritmos de balanceo de carga. Se analizará qué estrategias ofrecen menor latencia, mejor distribución de carga y mayor estabilidad. Asimismo, se evaluará el impacto de las medidas de seguridad en la protección contra ataques informáticos y la disponibilidad del sistema.

#### **5. Conclusiones y Recomendaciones**

Los hallazgos del estudio demostrarán la importancia de una arquitectura bien diseñada en entornos de computación en la nube. Se recomendarán estrategias óptimas para mejorar el rendimiento y la seguridad, alineadas con los estándares definidos por el NIST.

#### **6. Revisión Bibliográfica**

Para desarrollar este estudio, es fundamental revisar investigaciones previas y literatura académica que aborden los conceptos de sistemas distribuidos, balanceo de carga y seguridad en la nube. Algunos autores han explorado cómo las arquitecturas distribuidas pueden mejorar el rendimiento y la disponibilidad en entornos críticos.

Linthicum (2017) discute estrategias de nube empresarial, destacando la importancia de un diseño flexible y escalable. Por su parte, Murillo Rodríguez (2011) aborda la evolución de los sistemas distribuidos y su impacto en la optimización de recursos. Además, Suresh (2023) propone modelos de balanceo de carga inspirados en optimización bioinspirada para mejorar la distribución del tráfico.

- Linthicum, D. S. (2017). *Estrategia de nube empresarial*. O'Reilly Media.

- Murillo Rodríguez, J. M. (2011). *Sistemas distribuidos: Evolución e involución: Servidores, Cluster's, Grid Computing, Cloud Computing, Paralelismo y Concurrency*. Editorial UOC.
- Suresh, S. (2023). *Mitigación del balanceo de carga para la nube usando optimización de colonias de hormigas*. Springer.
- Universitat Oberta de Catalunya. (2017). *Introducción a la computación distribuida*. Editorial UOC.
- O'Reilly Media. (2020). *Una introducción a las bases de datos en la nube*. O'Reilly Media.
- Akamai Technologies. (2024). *¿Qué es el balanceo de carga en la nube?* Recuperado de <https://www.akamai.com>.