

Informe Técnico

Arquitectura Cloud para Sistema de Reservas

1. Introducción

El presente informe técnico describe el diseño de una arquitectura cloud para un sistema de reservas de espacios y recursos, desarrollado bajo un enfoque de microservicios. La solución propuesta busca cumplir con los requisitos de escalabilidad, seguridad, rendimiento y mantenibilidad definidos en el contexto del examen del módulo.

2. Objetivo del Proyecto

El objetivo principal es diseñar una arquitectura que permita gestionar reservas en tiempo real, adaptarse automáticamente a picos de demanda, proteger la información de los usuarios y facilitar futuras modificaciones sin afectar la estructura principal del sistema.

3. Descripción de la Arquitectura

La arquitectura propuesta se basa en microservicios, donde cada componente cumple una función específica y se comunica mediante interfaces bien definidas. Este enfoque permite el despliegue independiente, el mantenimiento eficiente y una alta escalabilidad.

Los principales componentes son:

- **API Gateway**, que actúa como punto único de entrada al sistema.
 - **Servicio de Autenticación**, encargado de la gestión de usuarios y control de acceso.
 - **Servicio de Reservas**, responsable de crear, consultar y cancelar reservas.
 - **Servicio de Recursos**, que administra la disponibilidad de espacios.
 - **Bases de datos independientes**, asociadas a cada microservicio.
 - **Sistema de monitoreo**, para el control de logs y métricas.
-

4. Escalabilidad y Rendimiento

La solución utiliza contenedores Docker para cada microservicio, permitiendo el escalado horizontal automático. Gracias al balanceo de carga y a la independencia de los servicios, el

sistema mantiene tiempos de respuesta adecuados incluso bajo altas cargas de usuarios concurrentes.

5. Seguridad

La seguridad del sistema se garantiza mediante autenticación y autorización con JWT, encriptación de datos en tránsito mediante HTTPS y la aplicación del principio de mínimo privilegio. Además, la separación de responsabilidades entre microservicios reduce el impacto de posibles fallos o vulnerabilidades.

6. Pruebas

Se implementan pruebas unitarias utilizando JUnit para validar el correcto funcionamiento de la lógica de negocio de cada microservicio. Asimismo, se realizan pruebas de carga y rendimiento para evaluar el comportamiento del sistema ante distintos volúmenes de usuarios.

7. Tecnologías Utilizadas

- Docker
 - Arquitectura de Microservicios
 - JUnit
 - JWT
 - Bases de datos relacionales y/o NoSQL
 - Principios del AWS Well-Architected Framework
-

8. Decisiones Arquitectónicas y Justificación

La elección de una arquitectura basada en microservicios se fundamenta en la necesidad de escalar de manera independiente cada componente del sistema, reducir el acoplamiento y facilitar la evolución funcional. A diferencia de una arquitectura monolítica, este enfoque permite aislar fallos, mejorar la resiliencia y desplegar cambios sin afectar al sistema completo.

El uso de contenedores Docker garantiza consistencia entre entornos y simplifica el despliegue. Asimismo, la incorporación de un API Gateway centraliza la seguridad y el control de acceso, mejorando la gobernanza del sistema.

9. Proyección y Mejoras Futuras

Como mejoras adicionales, se propone:

- Implementar mecanismos de cacheo para optimizar tiempos de respuesta.
- Incorporar observabilidad avanzada mediante métricas, trazabilidad distribuida y alertas.
- Aplicar patrones de resiliencia como circuit breaker y retry.
- Integrar pipelines CI/CD para automatizar pruebas y despliegues.

Estas mejoras permitirían aumentar la confiabilidad, disponibilidad y eficiencia operativa del sistema.

10. Conclusión

La arquitectura cloud propuesta no solo cumple con los requerimientos funcionales y técnicos establecidos en la consigna, sino que además incorpora criterios de diseño modernos orientados a la escalabilidad, seguridad y mantenibilidad. El enfoque adoptado entrega una solución optimizada, eficaz y preparada para evolucionar frente a nuevos desafíos y demandas del negocio, alineándose con las mejores prácticas de la industria en arquitectura de software en la nube.

Macarena Quijada Guzmán