

Es hora de que pongas en práctica todo lo aprendido. 

Este apartado tiene el objetivo de ayudarte a seguir potenciando tus habilidades, por lo que a continuación encontrarás diferentes **desafíos** que podrás resolver de forma independiente y a tu ritmo.

¡Manos a la obra!

1. Desafío

Consigna: Define escenarios de calidad específicos para un sistema de gestión de reservas en línea. Elige atributos de calidad clave, como seguridad, rendimiento y usabilidad, y crea tres escenarios de calidad para cada atributo, especificando estímulo, contexto, respuesta esperada y métrica de respuesta. Luego, diseña un diagrama básico de la arquitectura que soporte estos atributos de calidad.

Pasos:

1. Selecciona tres atributos de calidad prioritarios para el sistema y describe su relevancia.
2. Define tres escenarios de calidad para cada atributo, detallando estímulo, contexto, respuesta esperada y métrica.
3. Crea un diagrama arquitectónico básico, usando herramientas de diseño, que represente cómo el sistema soportará estos escenarios de calidad.

4. ¿Dónde se lleva a cabo?

Puedes realizar este desafío en una herramienta de diagramación como **Lucidchart** o **Draw.io** para crear el diseño arquitectónico. Para documentar los atributos de calidad y escenarios, puedes usar **Word**, **Google Docs** o **Notion**.

5. Tiempo de dedicación

2 Horas.

6. Recursos

- **Manual del módulo** para repasar conceptos sobre atributos y

escenarios de calidad.

- Referencia a la **ISO/IEC 25010**, que define los principales atributos de calidad en software, para asegurar una selección fundamentada de atributos.
- Guías en línea de diseño de escenarios de calidad, que ofrecen ejemplos de cómo detallar cada elemento de los escenarios (estímulo, contexto, respuesta esperada y métrica).

7. Plus +

Opcional: Para profundizar en el desafío, agrega un análisis de riesgos para cada escenario de calidad, señalando los puntos críticos y proponiendo soluciones de mitigación. Además, puedes investigar cómo marcos de calidad como **AWS Well-Architected Framework** pueden fortalecer la arquitectura para cumplir con estos escenarios de calidad.