

Arquitectura Microservicios

# Analisis de Arquitectura y Migracion a Microserviciós

---



**PONDERACIÓN: 1.5**

**Horas Aproximadas: 15**

Universidad San Carlos de Guatemala

Facultad de ingeniería.

Ingeniería en ciencias y sistemas

## Índice

<b>Índice</b>	<b>1</b>
<b>Competencias</b>	<b>2</b>
<b>Objetivos</b>	<b>2</b>
General	2
Específicos	2
<b>Descripción / Enunciado</b>	<b>2</b>
<b>Entregables</b>	<b>3</b>
<b>Consideraciones</b>	<b>3</b>
<b>Cronograma</b>	<b>3</b>
Ejemplo	3
<b>Evaluación</b>	<b>4</b>
<b>Valores</b>	<b>4</b>
<b>Referencias</b>	<b>4</b>

## Competencias

Capacidad para diseñar, implementar, escalar y mantener sistemas distribuidos basados en microservicios, asegurando la independencia, resiliencia, escalabilidad y mantenibilidad de cada servicio, de acuerdo con los principios de la arquitectura moderna y las necesidades del negocio. Objetivos

### General

- Aplicar los conocimientos adquiridos a lo largo de la carrera de Ingeniería en Ciencias y Sistemas para generar software de alta calidad y escalable, a través de diferentes técnicas de desarrollo y utilizando las últimas tecnologías.

### Específicos

- Que el estudiante aprenda a familiarizarse con la arquitectura de microservicios.
- Que el estudiante aprenda a diseñar su arquitectura.

## Descripción / Enunciado

En la empresa donde labora actualmente, el sistema encargado de la **gestión de finanzas corporativas** centraliza todos los procesos financieros: presupuestos, facturación, pagos a proveedores, conciliaciones bancarias y reportes de flujo de caja.

El sistema tiene las siguientes características:

- Cada módulo del sistema (autenticación, facturación, reportes financieros) se comunica con otros a través de un **bus central de servicios**.
- Los usuarios suben **archivos financieros CSV** que contienen facturas, gastos o ingresos.
- Antes de procesarse, los archivos deben pasar por un **proceso de validación con reglas internas**, verificando:
  - Que los montos sean correctos y coincidan con los presupuestos aprobados.
  - Que las fechas estén dentro del periodo contable vigente.
  - Que las categorías de gasto o ingreso sean válidas según la contabilidad de la empresa.

- Cada archivo o transacción requiere un **proceso de aprobación en 3 niveles**:
  1. **Nivel 1:** Revisión inicial por un asistente financiero, que verifica los datos básicos y adjunta comentarios si es necesario.
  2. **Nivel 2:** Validación por un supervisor o jefe de área, que revisa consistencia contable y cumplimiento de políticas internas.
  3. **Nivel 3:** Aprobación final por el director financiero o gerente de finanzas, quien autoriza que la transacción se ejecute.
    - Una vez aprobado en el último nivel, se envían automáticamente **notificaciones por correo** a los responsables de cada área financiera y, si aplica, a los proveedores.
    - Todos los eventos generados se registran en un **log centralizado**, que puede ser consultado para auditorías de seguridad y contables.
    - Se mantiene un **historial completo de archivos y transacciones**, disponible para consulta y descarga en cualquier momento.
    - Actualmente, el sistema puede presentar **problemas de rendimiento y escalabilidad** en cierres de mes o pagos masivos, debido a la manera en que los distintos módulos dependen del bus central para comunicarse.

### Caso Practico:

1. Analizar el escenario y **determinar qué tipo de arquitectura está utilizando**.
  - Elaborar un **diagrama de la arquitectura** identificada, mostrando los módulos, la comunicación entre ellos y los elementos clave del sistema.
  - Explicar de manera clara y fundamentada **por qué considera que esta es la arquitectura del sistema**, señalando las características que respaldan su conclusión.
2. Identificar los módulos, la manera en que se comunican y los cuellos de botella actuales.
3. **Rediseñar el sistema usando microservicios**, con cada funcionalidad independiente y escalable, comunicándose mediante APIs.
4. **Debe de incluir como documentacion alguna forma de poder entender y comprender las interacciones del sistema para poder verificar el flujo del proceso y la aplicación de las reglas de validacion solicitadas. (Se le recomienda la elaboracion de diagramas de secuencia para esto)**

## Entregables

Subir a UEDI el enlace del repositorio.

## Consideraciones

- Se debe hacer uso de un repositorio en la nube para realizar la entrega de su proyecto (Gitlab, Github, Bitbucket, etc.)
- El lenguaje a utilizar es de elección libre según lo requiera el estudiante. Aun así, se recomienda el uso de javascript/typescript o python.
- Se trabajará de manera individual.
- Las copias completas/parciales serán merecedoras de una nota de 0 puntos, los responsables serán reportados al catedrático de la sección y a la Escuela de Ciencias y Sistemas.
  - Nombre del Repositorio: **Practicas-SA-<<SECCIÓN>>-<<CARNE>>** o  
Crear carpeta dentro del repositorio con el nombre **P3** e incluir la documentación dentro de ella.
  - Agregar al auxiliar al repositorio, con el rol Developer:
    - o Sección A: **hkjvasquez**
    - o Sección B: **di3gini**

## Cronograma

El cronograma permite planificar y organizar las diferentes etapas del desarrollo de la práctica

Ejemplo

Tarea	Fecha
Asignación practica	18 agosto 2025
Fecha de entrega	23 agosto 2025
Fecha de calificación	24-25 agosto 2025

## Rúbrica de Calificación

La evaluación de la práctica busca medir el cumplimiento de los objetivos planteados, así como la correcta aplicación de los conocimientos técnicos y habilidades.

Se debe agregar una tabla con los aspectos a calificar con su respectiva puntuación.


Descripción	Valor
<b>Diseño de Microservicios</b>	—
Separación correcta de microservicios	10
No aplicar anti-patrones de microservicios	10
Bases de datos correctamente separadas	5
Aplicación de herramientas y estrategias de microservicios	10
Reflejo de las reglas de validacion del negocio	5

<b>Documentación</b>	—
Diagrama de Arquitectura	10
Descripción de la solución	10
Diagrama ER de las bases de datos	10
Descripción de la solución	10
Reflejo de las reglas de validacion del negocio	10
<b>Preguntas</b>	—
Pregunta 1	5
Pregunta 2	5

## Valores

En el desarrollo de la práctica, se espera que cada estudiante demuestre honestidad académica y profesionalismo. Por lo tanto, se establecen los siguientes principios:

1. **Originalidad del Trabajo**
  - Cada estudiante o equipo debe desarrollar su propio código y/o documentación, aplicando los conocimientos adquiridos en el curso.
2. **Prohibición de Copias y Plagio**
  - Si se detecta la copia total o parcial del código, documentación o cualquier otro entregable, la calificación será de **0 puntos**.
  - Esto incluye la reproducción de código entre compañeros, la reutilización de proyectos de semestres anteriores o el uso de código externo sin la debida referencia.
3. **Uso Responsable de Recursos Externos**
  - El uso de bibliotecas, frameworks y ejemplos de código externos está permitido, siempre y cuando se referencian correctamente y se comprendan plenamente. ( Consultar con el catedrático su política)
4. **Revisión y Detección de Plagio**
  - Se podrán utilizar herramientas automatizadas y revisiones manuales para identificar similitudes en los proyectos.
  - En caso de sospecha, el estudiante deberá justificar su código y demostrar su desarrollo individual o en equipo. Si este extremo no es comprobable la calificación será de **0 puntos**.

 Al detectarse estos aspectos se informará al catedrático del curso quien realizará las acciones que considere oportunas.