

# Sistemas Distribuidos y Microservicios

Héctor Xavier Limón Riaño  
email: [hlimon@uv.mx](mailto:hlimon@uv.mx)

# Quien soy

- ▶ Docente de tiempo completo de la facultad de Estadística e Informática, Universidad Veracruzana
- ▶ Adscrito al programa educativo de Redes y Servicios de Cómputo
- ▶ Licenciado en informática, maestría y doctorado en IA
- ▶ Estancias de Investigación en la universidad de Valencia y Bologna
- ▶ Miembro del Sistema Nacional de Investigadores

# Áreas de interés

- ▶ Sistemas multi-agente
- ▶ Minerería de datos
- ▶ Ciberseguridad
- ▶ Sistemas distribuidos
- ▶ Ingeniería de Software

# Investigación relacionada a microservicios

- ▶ Patrones de diseño
- ▶ Seguridad en microservicios
- ▶ Despliegue de microservicios
- ▶ Manejo de transacciones en microservicios
- ▶ Deuda técnica asociada a microservicios

# Introducción

- ▶ Vivimos en un mundo conectado a través de servicios de red
- ▶ Necesidades y usos en todos los niveles:
  - ▶ Educación
  - ▶ Gobierno
  - ▶ Ciencia
  - ▶ Industria
  - ▶ Instituciones financieras

- ▶ Tendencias en servicios de red:
  - ▶ Entretenimiento
  - ▶ Aplicaciones bancarias
  - ▶ Comercio electrónico
  - ▶ IOT
  - ▶ Análisis de datos
  - ▶ Servicios en la nube

- ▶ Propiedades que deben tener estos sistemas:
  - ▶ Escalabilidad
  - ▶ Mantenibilidad
  - ▶ Seguridad
  - ▶ Facilidad de despliegue

# Sistemas distribuidos

- ▶ Cualquier sistema cuyas partes se encuentren en más de una computadora, posiblemente con separación geográfica
- ▶ Motivación principal: compartir recursos
- ▶ Beneficios:
  - ▶ Escalabilidad
  - ▶ Tolerancia a fallas (redundancia)
  - ▶ Disponibilidad



- ▶ Retos
  - ▶ Aumento de complejidad (diseño, despliegue, persistencia, recuperación de errores, etc.)
  - ▶ Mayor superficie de ataque
  - ▶ Costo (monetario y de mantenimiento)

- ▶ Service Oriented Architecture
- ▶ Aproximación para diseñar sistemas distribuidos
- ▶ Representar sistemas distribuidos como un conjunto de servicios
- ▶ Modelo popularizado por Servicios Web

# Arquitectura de Microservicios

- ▶ Arquitectura de software inspirada en SOA y metodologías de desarrollo ágiles
- ▶ Surge alrededor del 2014
- ▶ Surge como respuesta a las de necesidades de extensibilidad y escalabilidad de los sistemas distribuidos modernos
- ▶ Así como para lidiar con los rápidos cambios tecnológicos y los cada vez más cortos ciclos de entrega de software

# Arquitectura de microservicios

- ▶ La idea es descomponer el sistema distribuido en servicios pequeños y altamente cohesivos, con un mínimo control centralizado
- ▶ La comunicación entre microservicios se realiza a través de mecanismos ligeros y estandarizados
- ▶ La arquitectura facilita la heterogeneidad tecnológica
- ▶ Cada microservicio tiene su propio ciclo de vida

# Ventajas de MSA

- ▶ Facilita la división de trabajo en los equipos de desarrollo
- ▶ Favorece el bajo acoplamiento entre servicios
- ▶ Permite una integración simple de nuevas tecnologías
- ▶ Facilita el despliegue de nuevas versiones
- ▶ Facilita la escalabilidad
- ▶ El tiempo de mantenimiento se reduce
- ▶ Facilita la prueba de software

- ▶ Seguridad
  - ▶ Mayor superficie de ataque
  - ▶ La heterogeneidad conlleva a más componentes con vulnerabilidades conocidas
  - ▶ La autenticación y autorización entre microservicios es compleja
  - ▶ Necesidad de proteger cada end-point

- ▶ Persistencia
  - ▶ El manejo de transacciones que involucran a varios microservicios es compleja y requiere aproximaciones especiales
  - ▶ La heterogeneidad en modelos de datos complica operaciones que involucran a más de una fuente de datos
  - ▶ Complejidad potenciada para mantener integridad y consistencia

- ▶ Organizacionales
  - ▶ Segmentar equipos de desarrollo dificulta la comunicación entre ellos
  - ▶ Competencia por recursos de infraestructura entre equipos de desarrollo
  - ▶ Diversificación técnica
  - ▶ Aumento de costos de la infraestructura



- ▶ Manejo de fallos
  - ▶ Muchas oportunidades para que los componentes fallen
  - ▶ Es difícil detectar que una instancia de microservicio falló
  - ▶ El monitoreo es complejo

- ▶ Diseño
  - ▶ Es difícil establecer la granularidad de cada microservicio
  - ▶ No hay estándares de modelado o metodológicos específicos
  - ▶ Hay brechas importantes entre academia e industria

# Enfrentar los retos

- ▶ Enfoque en el diseño
- ▶ Uso de patrones y tácticas
- ▶ Seguimiento de mejores prácticas

**¡Gracias!**