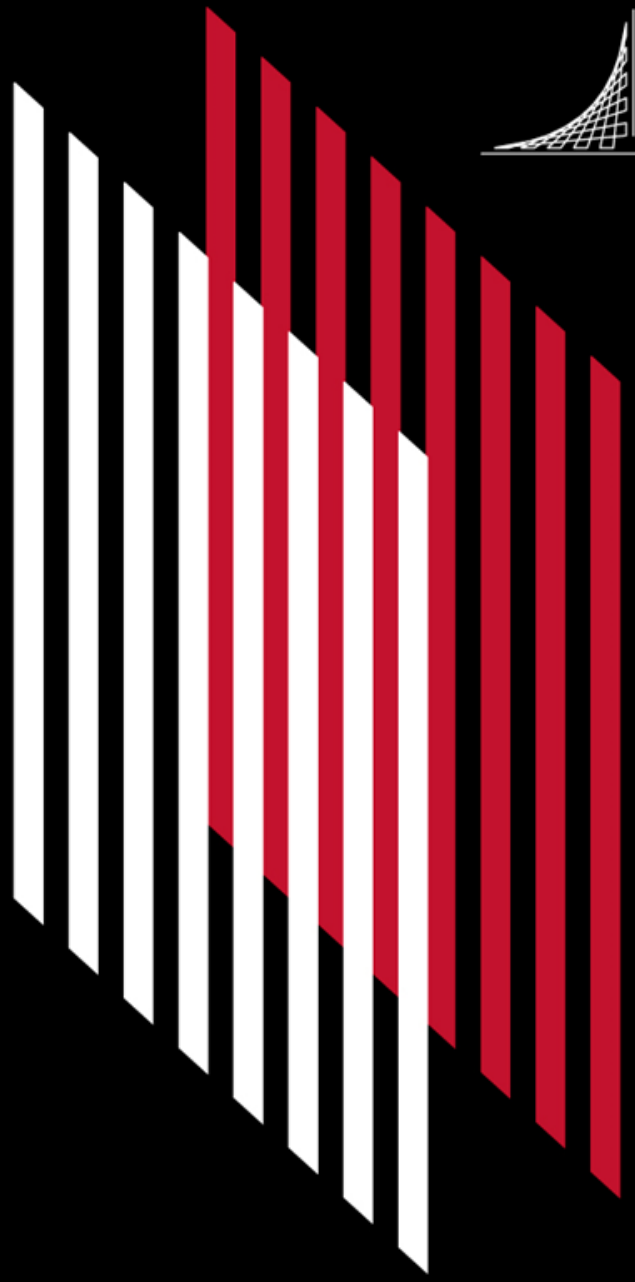
Migrando Arquitecturas Antiguas a Microservicios:

Una Cuestión de Escalabilidad y Mantenibilidad.



Presentado por:

- Iván Camilo Rincón Saavedra.
- Laura Valentina García León.
- Juan David Murillo Giraldo.
- Leonardo Galeano Garzón.

Arquitecturas empresariales (AREP)

2022 - 1

Bogotá D.C

Objetivos

- Estudiar y analizar arquitecturas modernas con el fin de entender cómo las nuevas tendencias arquitectónicas mejoran el rendimiento de un sistema.

- Identificar qué factores son decisivos para saber en qué momento es necesario escalar un sistema de información.

Problema

En los últimos años, la gran mayoría de sistemas han sufrido un crecimiento exponencial de usuarios.

Ocasionando un desborde en la capacidad de estos para manejarlos y responder.

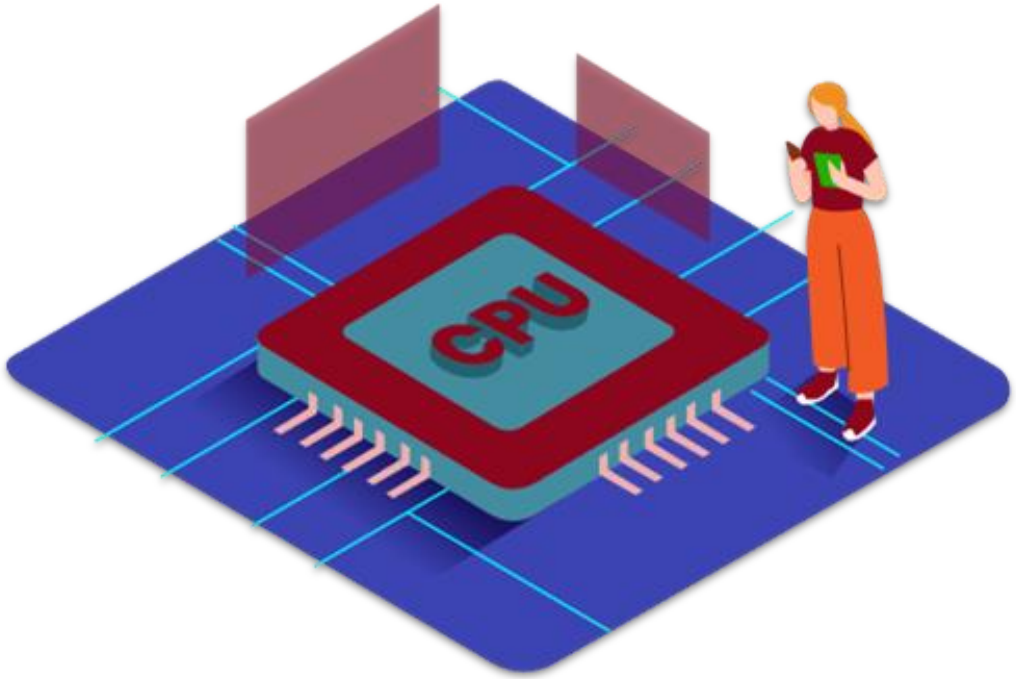
Generando un aumento de complejidad en la captura, gestión, procesamiento y análisis de datos.

Sistemas Monolíticos

Son aquellos en los que en un único repositorio se almacena todo el código relacionado a las funcionalidades y servicios del negocio.

Es decir tanto el el backend como el frontend comparten recursos en su almacenamiento.

Esto genera un alto acoplamiento y la complejidad de la lógica de negocio.

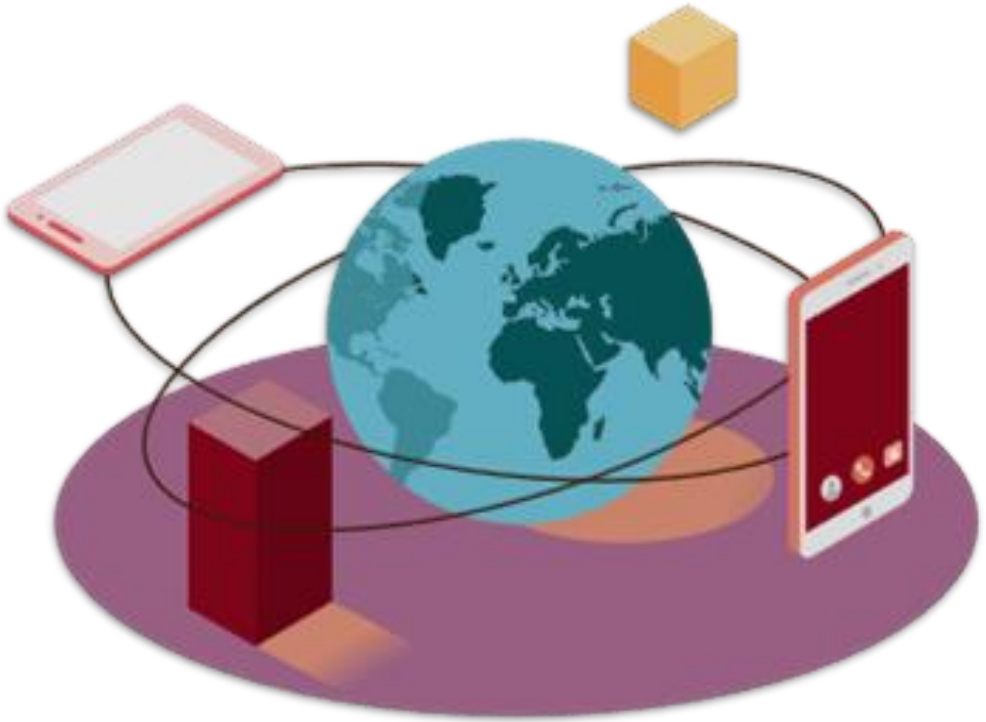


¿Microservicios como Arquitectura?

Planteado como un estilo arquitectónico, permite diseñar aplicaciones o sistemas de forma atómica.

Busca generar **“pequeños”** componentes que exploten las funcionalidades principales de la aplicación o sistema.

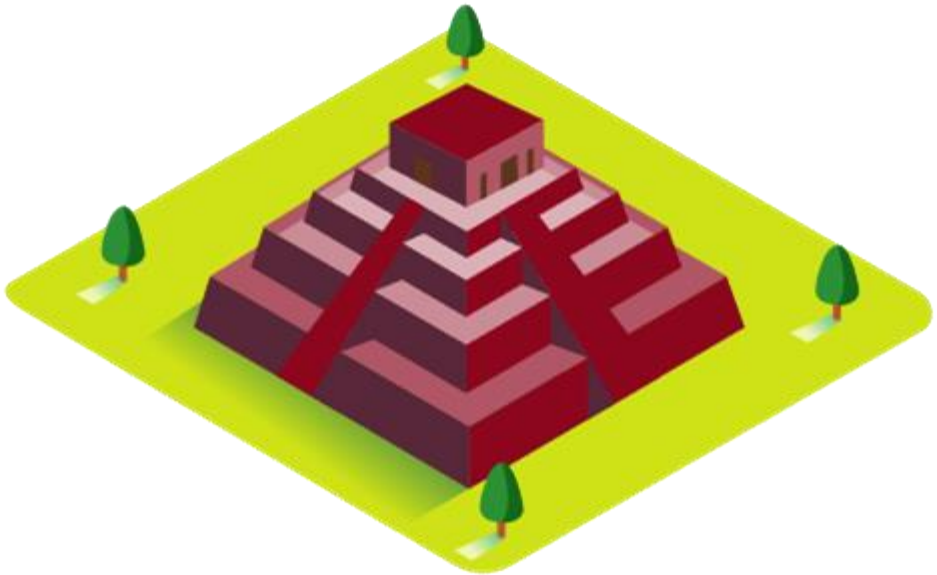
Dichos componentes deben ser ejecutados y funcionar de manera independiente, siendo capaces de comunicarse entre sí.



Evolución Arquitectónica

Pasado:

Las aplicaciones solían manejar una arquitectura monolítica, dejando en una base de código única todos los recursos necesarios para el servicio.



Pros:

- ★ Tiempos de respuesta bajos.
- ★ Fácil de desarrollar y desplegar.

Contras:

- ★ Mantenimiento complejo.
- ★ Baja disponibilidad en actualizaciones.
- ★ Difícil escalabilidad.

Evolución Arquitectónica

Presente:

Se maneja un frontend construido como monolito, pero que aprovecha una galería de API's, en donde cada uno de estos es independiente de los demás y funciona por sí mismo.

Pros:

- ★ Fácil mantenimiento.
- ★ Alta disponibilidad.
- ★ Escalabilidad sencilla.

Contras:

- ★ Desarrollo complejo.
- ★ Integración de datos masivos compleja.

Evolución Arquitectónica

Futuro:

Se espera que el comportamiento de microservicios sea aplicable tanto para Backend como para Frontend.

Pros:

- ★ Mantenimiento sencillo.
- ★ Alta disponibilidad.

Contras:

- ★ Desarrollo excesivamente complejo.
- ★ Costo de infraestructura altos.
- ★ Mantenimiento y seguimiento constante.

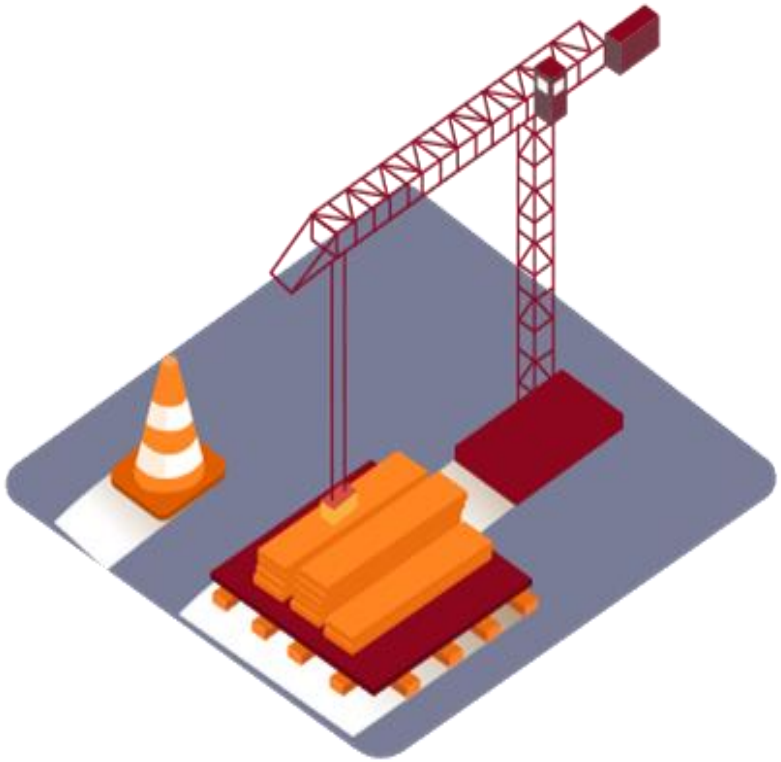
Beneficios de una arquitectura moderna

- Flexibilidad
- Modularidad
- Disponibilidad
- Mantenibilidad
- Fiabilidad
- Confianza y seguridad

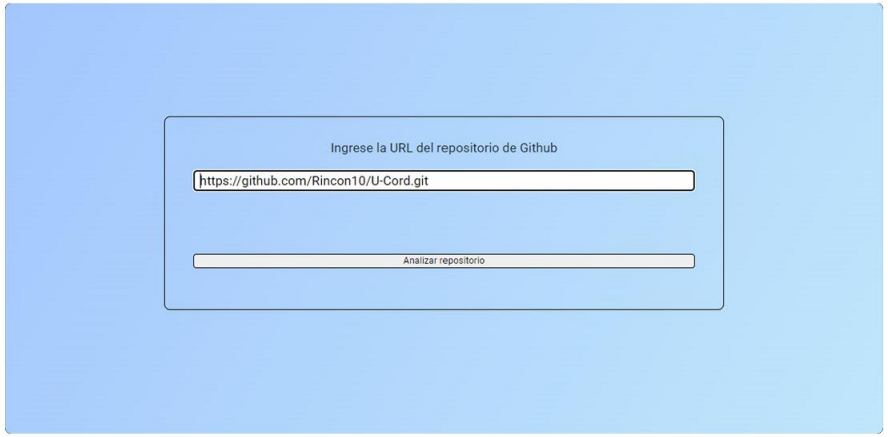
Prototipo

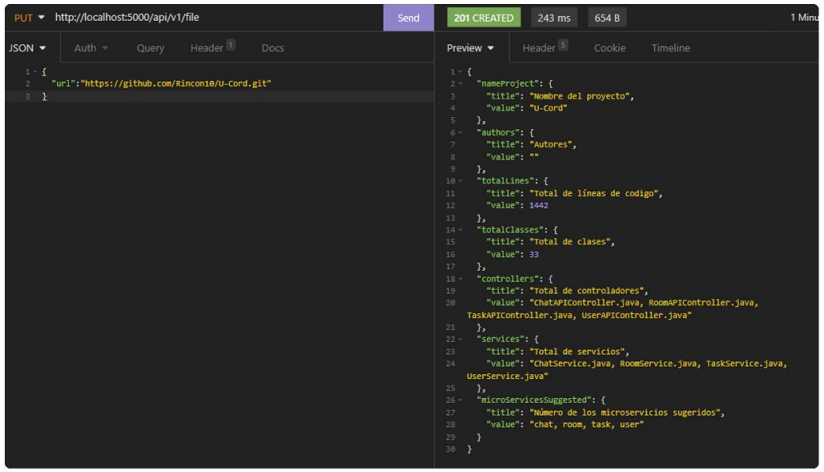
- Se propone el desarrollo de una herramienta que permita el análisis de código estático de un proyecto, principalmente escrito en Java.

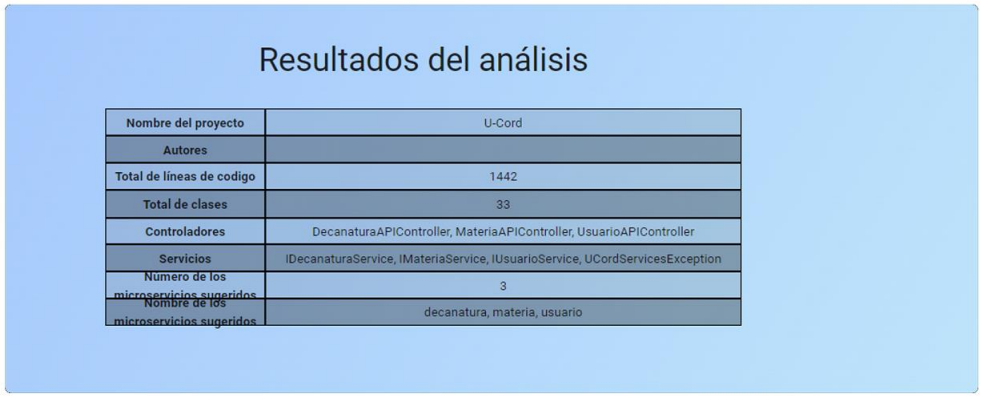
- El cual busca generar insights de valor que le permitan a los programadores conocer que tan viable y sencilla es la migración a microservicios.



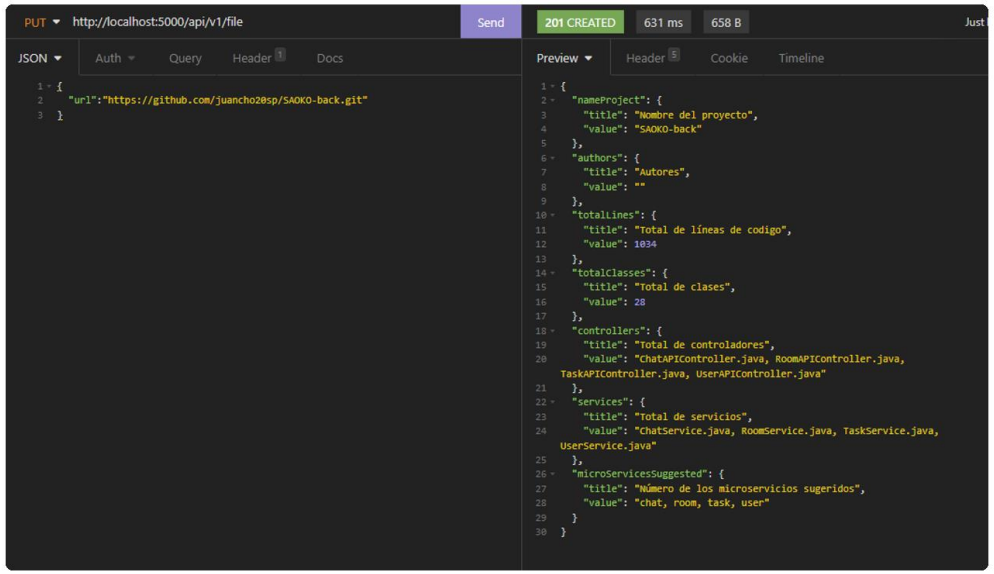
Prototipo-ejemplo01







Prototipo-ejemplo02

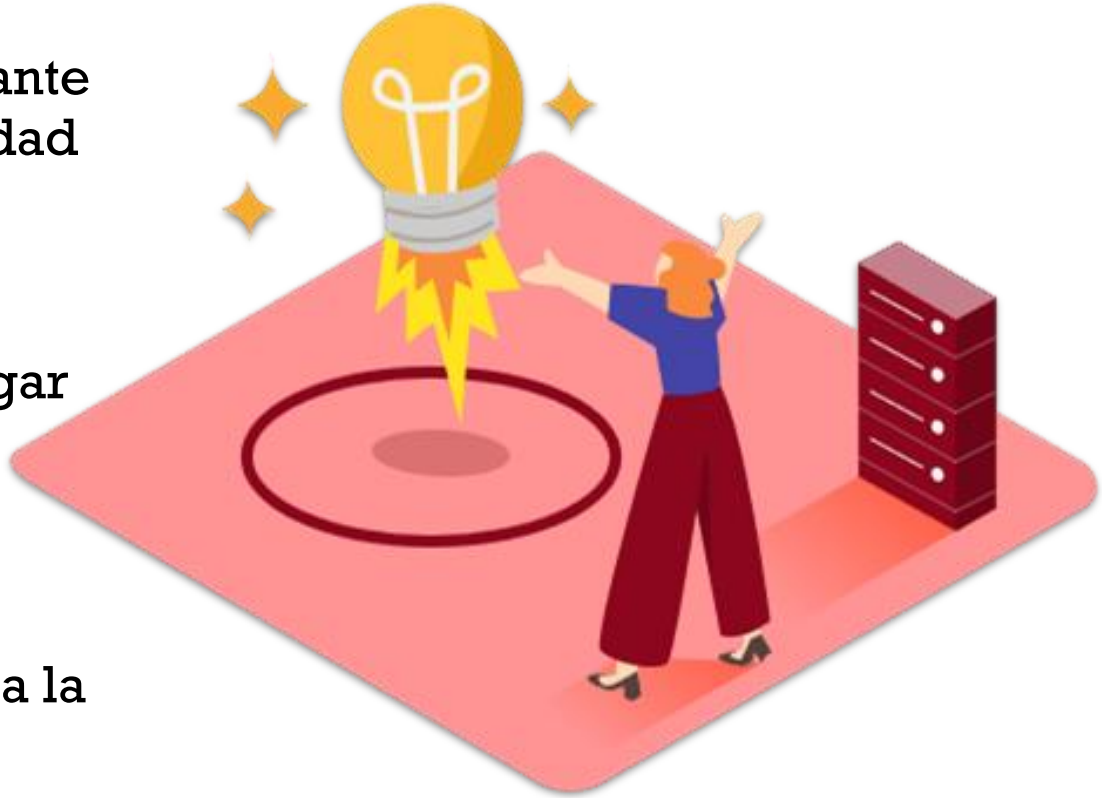


Resultados del análisis

Nombre del proyecto	SAOKO-back
Autores	
Total de líneas de codigo	1034
Total de clases	28
Controladores	ChatAPIController, RoomAPIController, TaskAPIController, UserAPIController
Servicios	ChatService, RoomService, TaskService, UserService
Numero de los microservicios sugeridos	4
Nombre de los microservicios sugeridos	chat, room, task, user

Conclusiones.

- Los microservicios son una alternativa interesante cuando se presentan problemas de escalabilidad y mantenibilidad dentro de un proyecto.
- Los microservicios permiten tercerizar o delegar actividades de mantenimiento y desarrollo a diferentes equipos.
- La migración de una aplicación monolítica para la arquitectura de microservicios es compleja.



Referencias

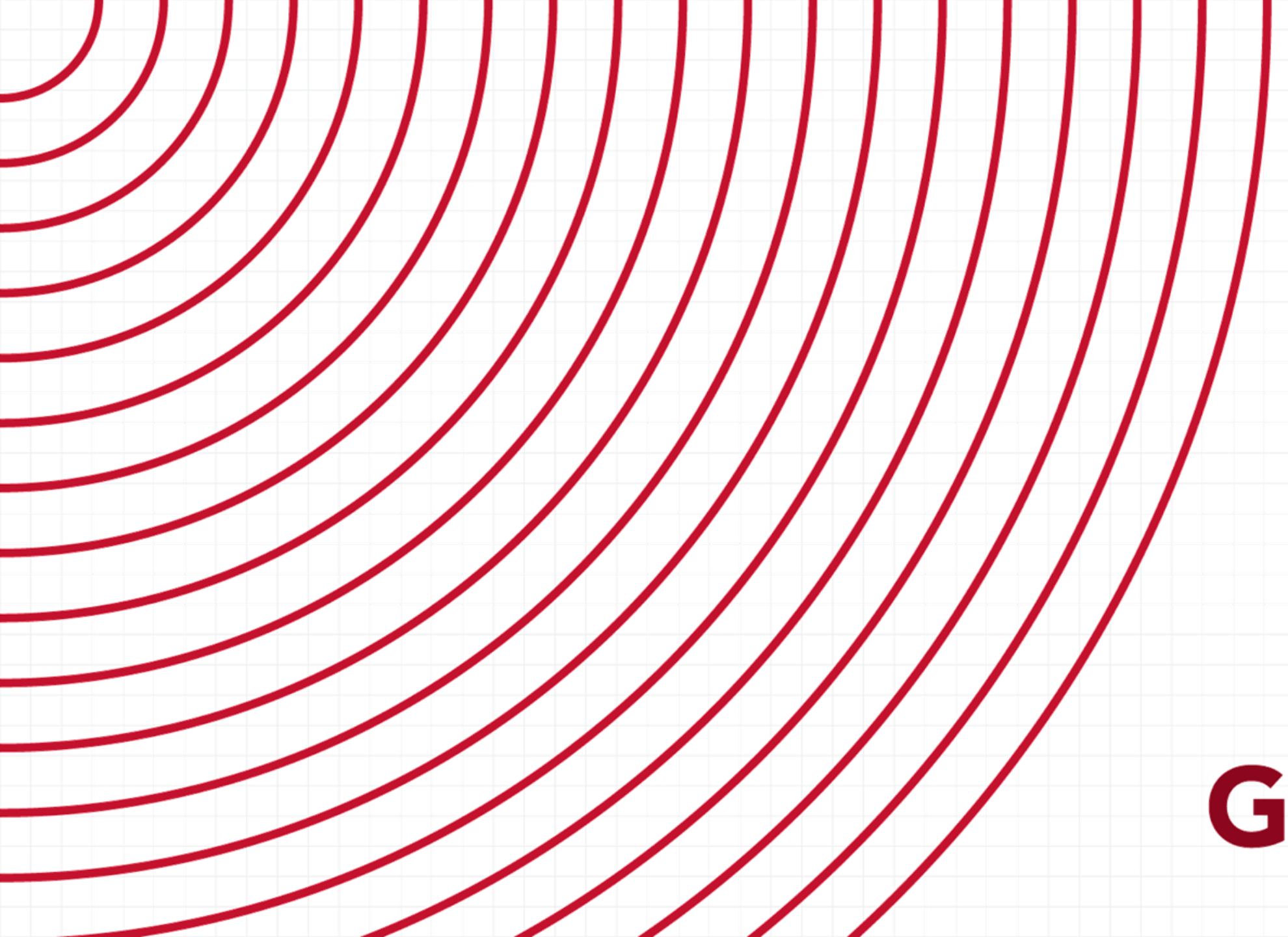
- Ponce Mella, Francisco Márquez, Gastón Astudillo, Hernán. (2019).Migrating from monolithic architecture to microservices: A Rapid Review.

- Monoliths-into-microservices (2022).Retrieved 13 May 2022, from https://docs.aws.amazon.com/es_es/prescriptive-guidance/latest/patterns/decompose-monoliths-into-microservices-by-using-cqrs-and-event-sourcing.html.

Referencias

- Migrando una aplicación monolítica para la arquitectura serverless y microservicios (2022). Retrieved 13 May 2022, from https://aws.amazon.com/es/blogs/aws-spanish/migrando-una-aplicacion-monolitica-para-la-arquitectura-serverless-y-microservicios

- F. Ponce, G. Márquez and H. Astudillo, ”Migrating from monolithic architecture to microservices: A Rapid Review,” 2019 38th International Conference of the Chilean Computer Science Society (SCCC), 2019



ESCUELA
COLOMBIANA
DE INGENIERÍA
JULIO GARAVITO

Gracias