

DIAGRAMAS C4

GESTIÓN DEL DESARROLLO DE SOFTWARE
PROF. MANGO EDUARDO

Diagramas de Arquitectura de Software

Los **Diagramas de Arquitectura de Software** son representaciones visuales que muestran cómo se estructura un sistema de software.

Se encargan de mostrar:

- **Componentes** principales del sistema
- **Relaciones** entre componentes y con sistemas externos
- **Flujos de datos** dentro del sistema

Diagramar la arquitectura de nuestro software nos trae varias ventajas, entre otras:

- Los diagramas de arquitectura proporcionan una **visión clara** de los componentes y la estructura del sistema.
- Facilita la **comunicación** durante el proceso de diseño
- Permiten **mantener y escalar** los sistemas con mayor facilidad
- Permite **detectar errores** como lógica defectuosa o suposiciones incorrectas.

Modelo C4

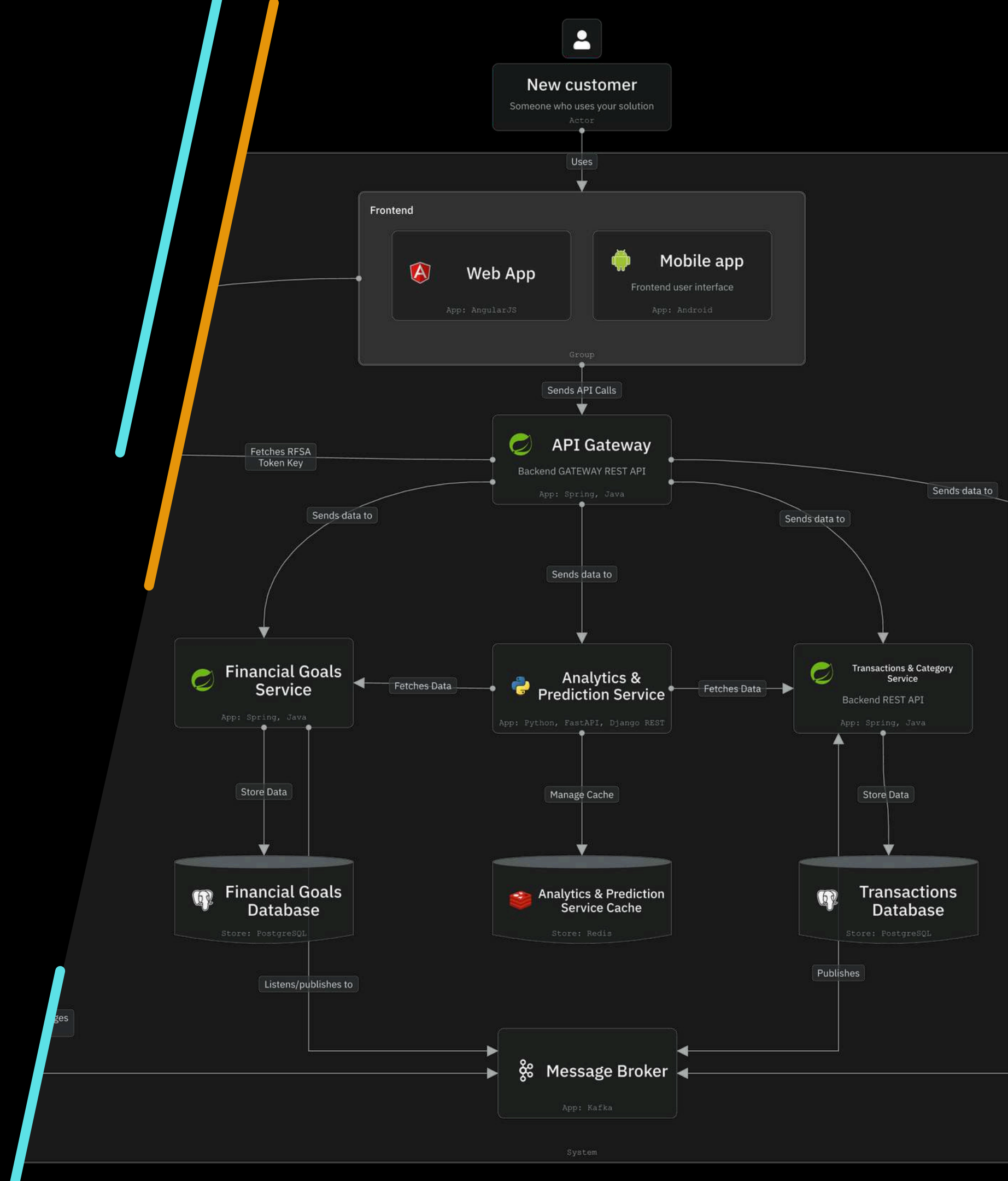
El modelo C4 es un **modelo de diagramación de arquitectura de software agnóstico a arquitectura y paradigma**, creado por Simon Brown. Propone 4 niveles de diagramas para describir un sistema con distintos grados de detalle. Estos niveles son:

Contexto: Representa cómo nuestro sistema se comunica con el usuario y con sistemas externos.

Contenedores: Representan los grandes contenedores lógicos como APIs, bases de datos y cómo estos se comunican entre sí.

Componentes: Represente los componentes o módulos que componen un contenedor.

Código: Muestra detalles de implementación del código, como clases, interfaces, etc. Es similar a un UML



ABSTRACCIONES



PERSONA

**REPRESENTA AL
USUARIO DEL SISTEMA**



CONTENEDOR

**REPRESENTA
APLICACIONES Y
ALMACENES DE DATOS
INDEPENDIENTEMENTE
EJECUTABLES.**



SISTEMA

**REPRESENTA EL NIVEL
MAS ALTO DE
ABSTRACCIÓN**



COMPONENTE

**BLOQUES Y MODULOS
LOGICOS QUE
COMPONEN A LOS
CONTENEDORES.**

DIAGRAMA DE CONTEXTO

El diagrama de **contexto, de nivel 1 o C1**, es aquel que muestra el sistema de software que está construyendo y **cómo encaja en el mundo** en términos de las personas que lo utilizan y los otros sistemas de software con los que interactúa.

Es el punto de entrada a nuestra arquitectura y el nivel más alto de abstracción.



DIAGRAMA DE CONTEXTO

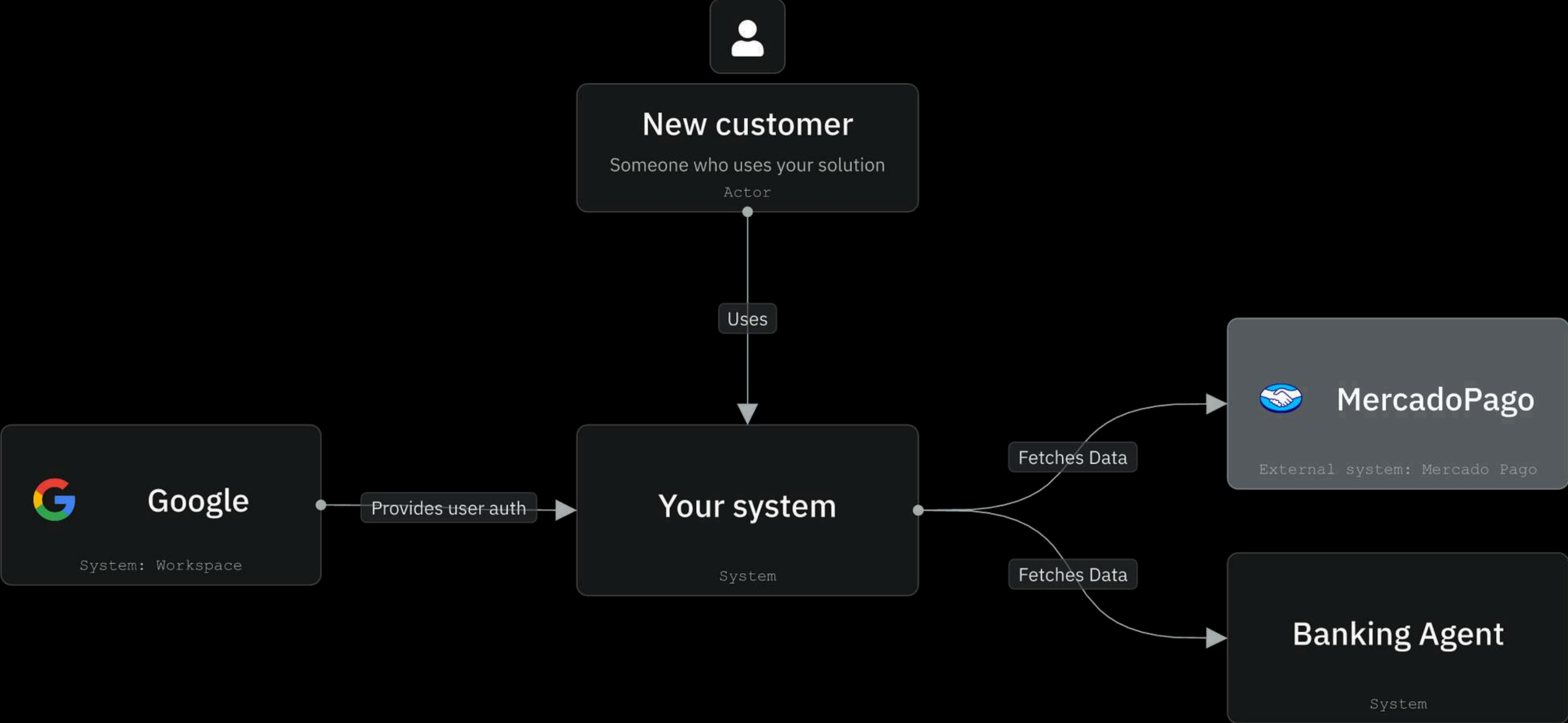


DIAGRAMA DE CONTENEDORES

El nivel 2 (C2), un **diagrama de contenedor**, amplía el sistema de software y muestra los contenedores (aplicaciones, almacenamiento de datos, microservicios, etc.) que componen este sistema de software. Aquí ya se suelen tomar e incluir decisiones tecnológicas.

Pueden imaginar el diagrama C2 como un **“zoom”** a nuestro sistema del diagrama C1, por lo que es clave **mantener las relaciones previamente planteadas.**



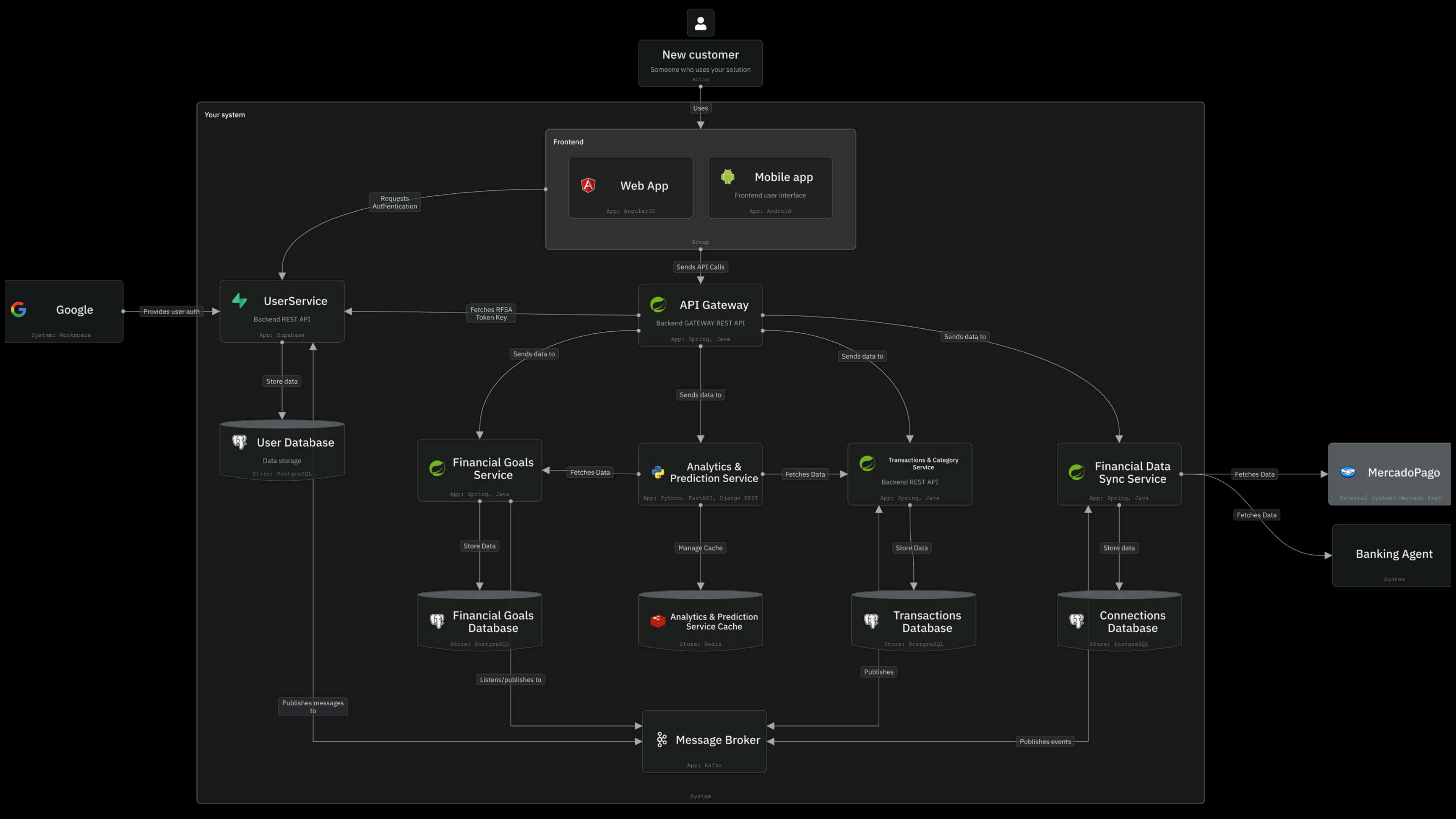
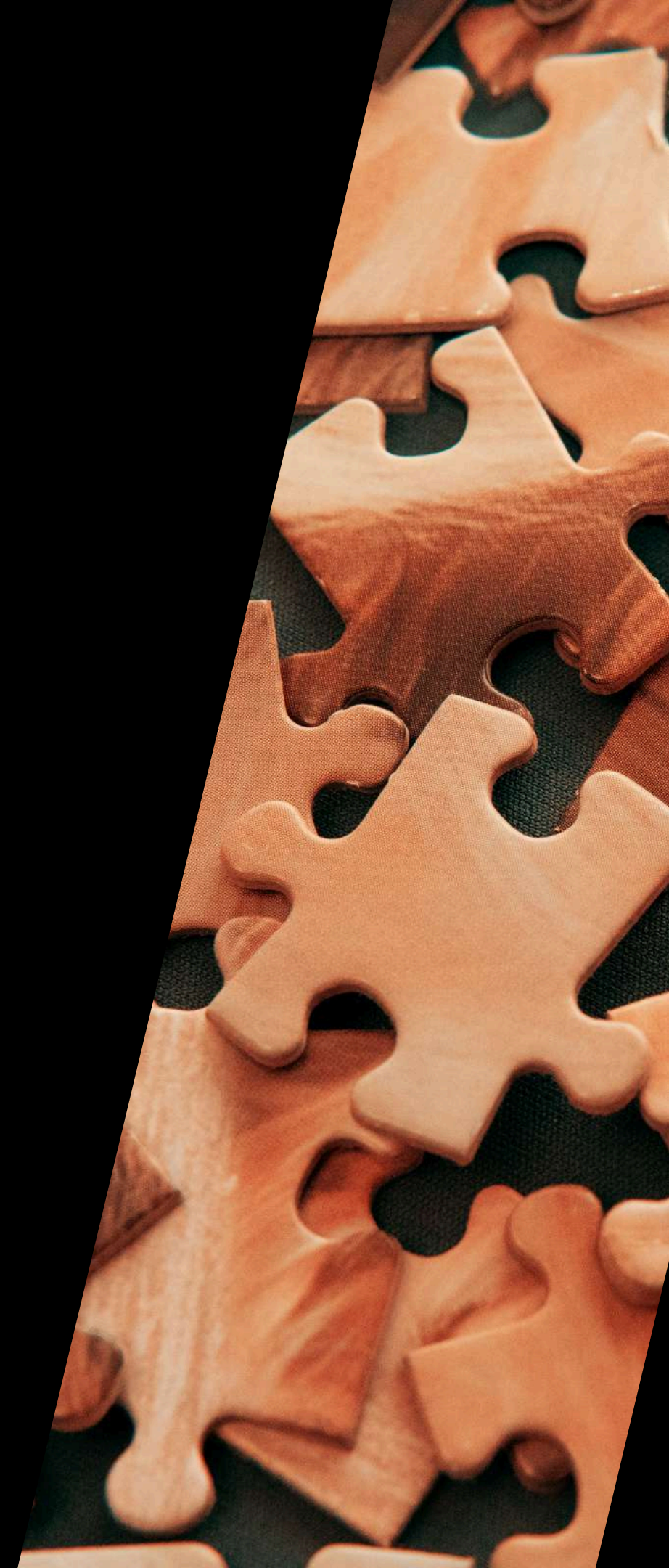


DIAGRAMA DE COMPONENTES

El nivel 3 (C3), un **diagrama de componentes**, expande un contenedor individual para mostrar los componentes que contiene. Estos componentes deben asignarse a **abstracciones reales** (por ejemplo, una agrupación de códigos) en función de su código.

Esta etapa trata sobre entender cómo **interactúan estos componentes**, un factor crítico para asegurar la fortaleza y capacidad de respuesta general del sistema.



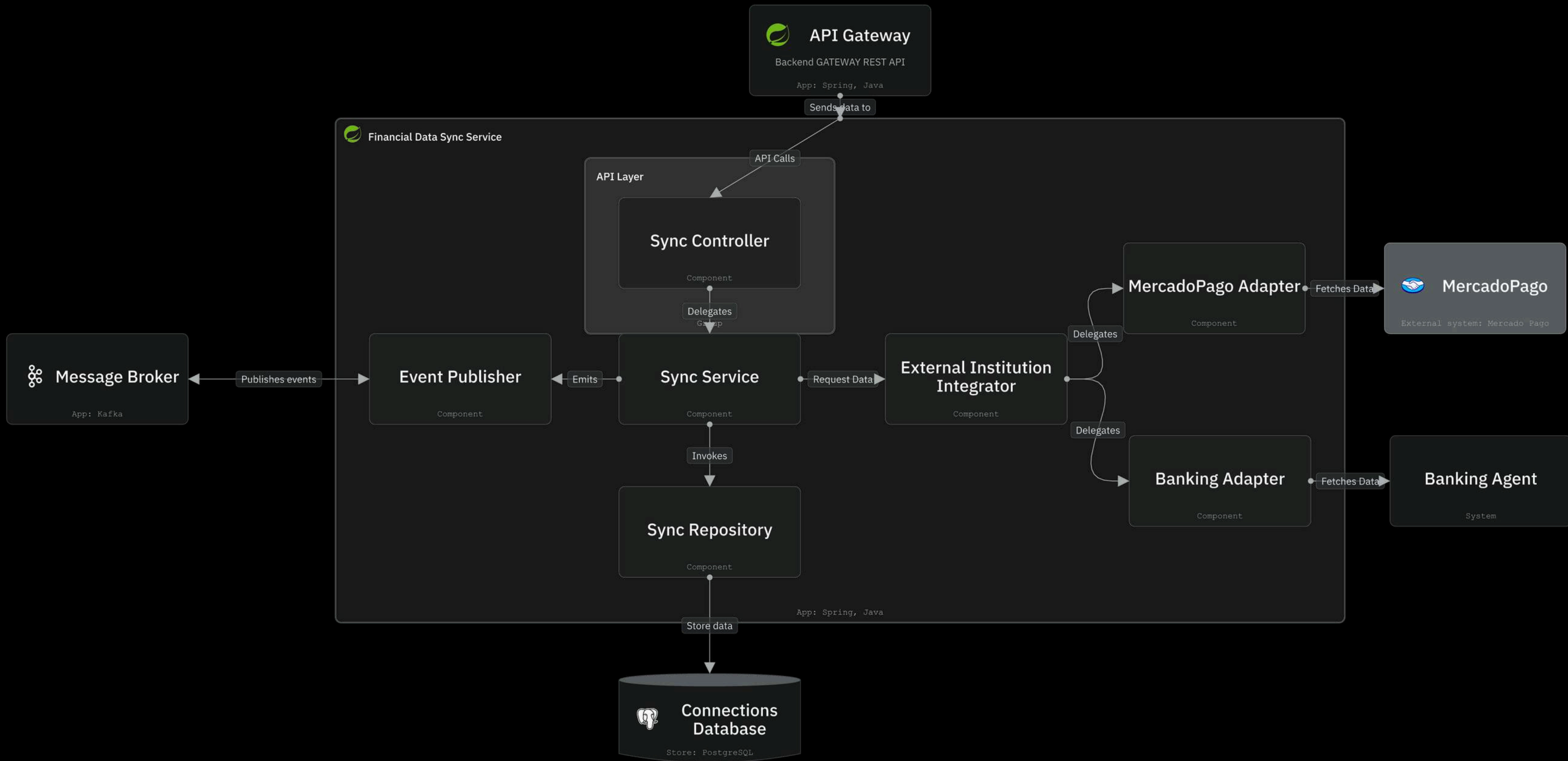


DIAGRAMA DE CODIGO

El **diagrama de código** (C4) proporciona una vista detallada de los **bloques de construcción del sistema**, mostrando las clases e interfaces que sustentan los componentes. Es una inmersión profunda en la anatomía del código, resaltando las relaciones y dependencias que son fundamentales para la estabilidad y rendimiento del sistema.

Para esto se suelen usar herramientas ya existentes como **diagramas UML**.





Mejores prácticas

Consistencia

Al crear diagramas del modelo C4, es crucial utilizar símbolos y notaciones uniformes.

Nivel de detalle y abstracción

Demasiado detalle puede convertir tu diagrama en un laberinto complejo que es difícil de descifrar. Por otro lado, demasiado poco detalle puede causar confusión o forzar suposiciones.

Enfoque colaborativo

El diagrama C4 es una herramienta de comunicación que debe construirse con aportes de toda la organización. Fomenta la colaboración involucrando a diferentes interesados y miembros del equipo en el proceso de creación.

Documentación y notación

Una buena documentación y anotaciones claras son la base de un modelo C4 efectivo. Proporcionan contexto y claridad.

DIAGRAMAS C4

GESTIÓN DEL DESARROLLO DE SOFTWARE
PROF. MANGO EDUARDO

Información obtenida de:
<https://c4model.info/#app>
<https://c4model.com/>

Universidad Tecnológica Nacional de Mar del Plata

