

# Implementación inicial de APIs simuladas

# Los Dos Sistemas Principales

---



## Sistema CRM Básico

Gestión de relaciones con clientes,  
seguimiento de pedidos y datos de  
contacto.



## Gestión de IoT Básico

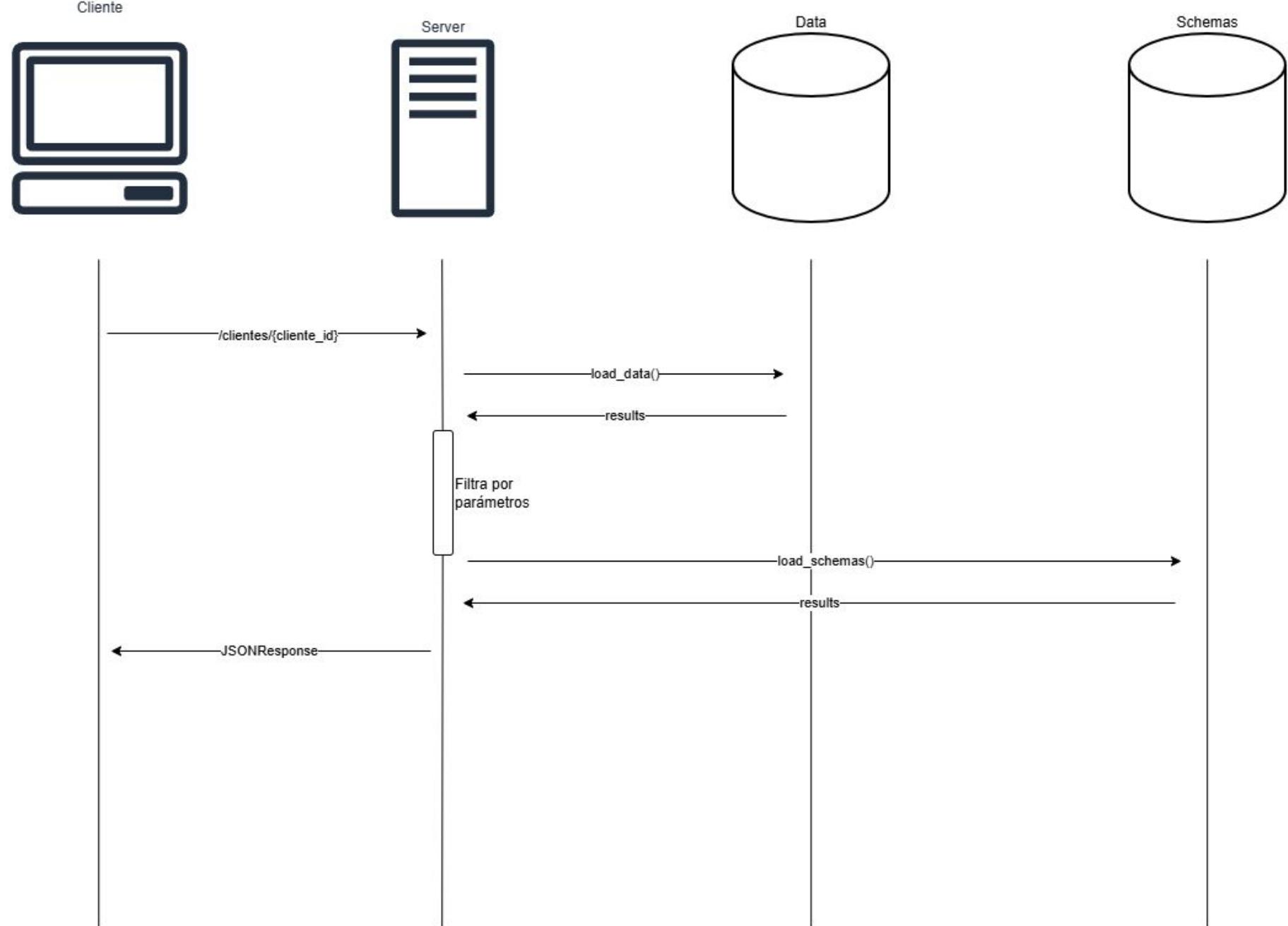
Recolección de datos de sensores y  
monitoreo de lecturas de dispositivos.



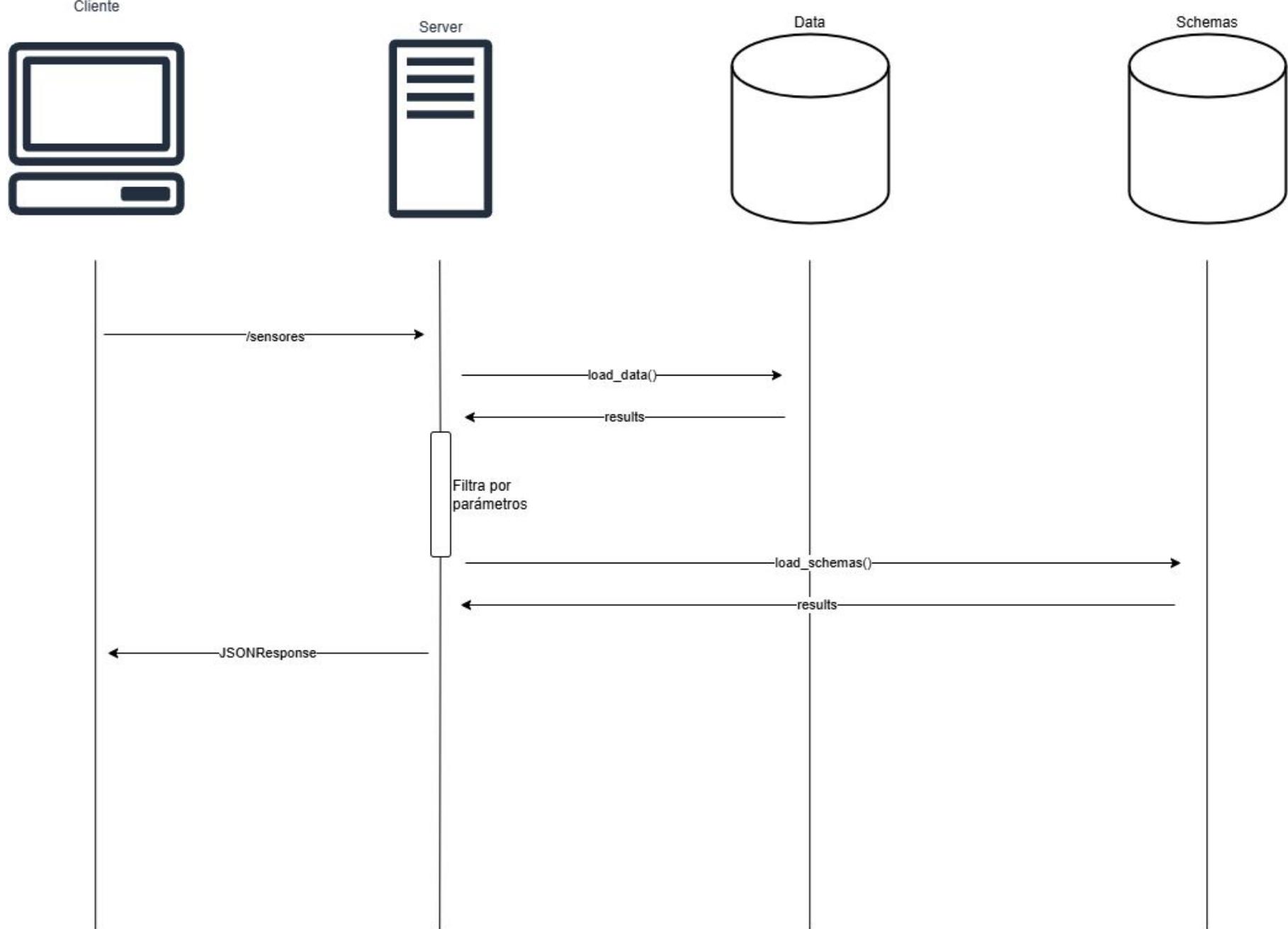
## Doble Stack

Ambos sistemas están disponibles en  
Python y JavaScript para diversas  
arquitecturas.

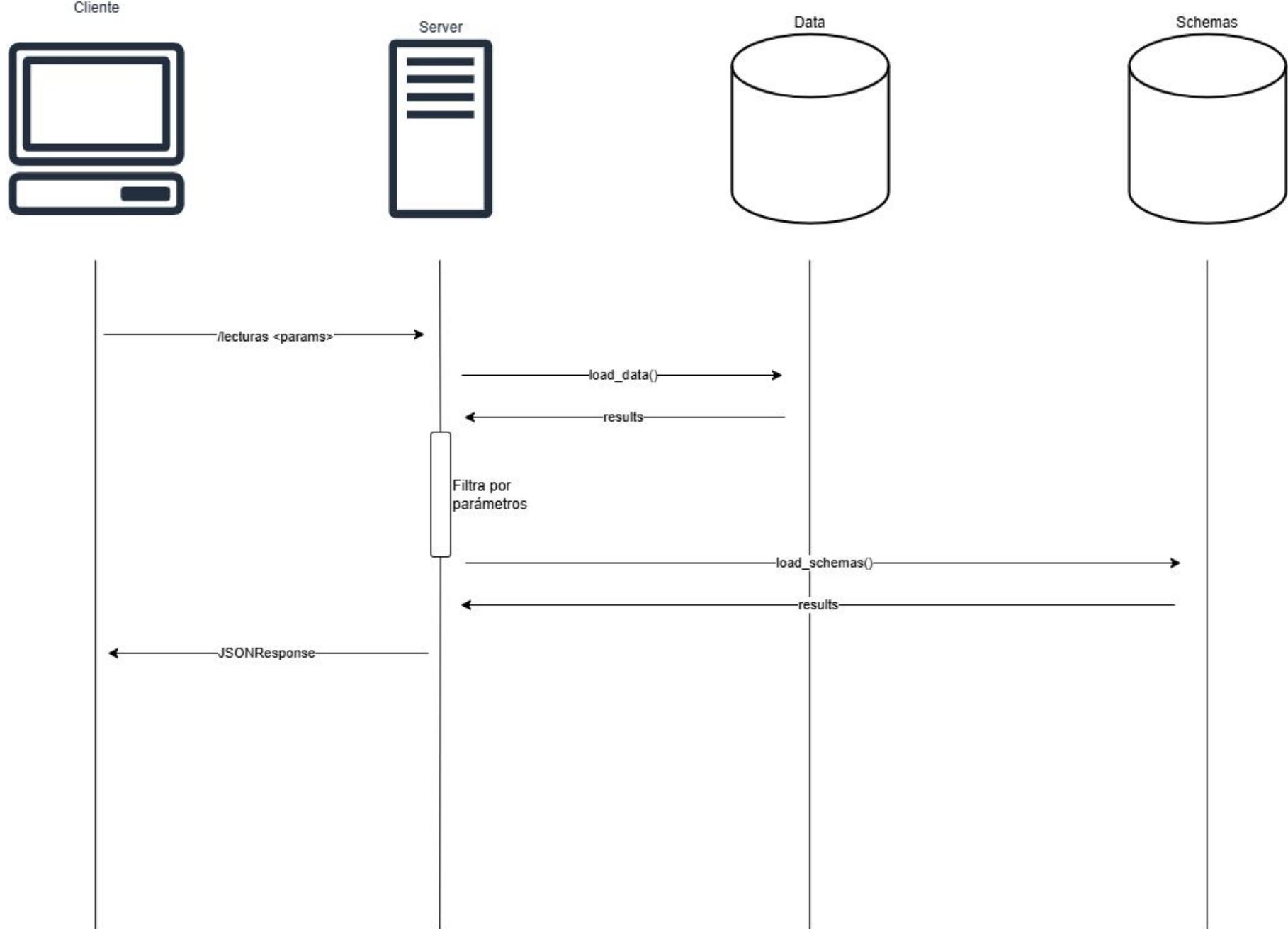
# Flujo 1 CRM Clientes



# Flujo 2.1 IoT Sensores



# Flujo 2.2 lot Lecturas



# Estructura de Archivos (TuGranjita)

---

```
TuGranjita
| .gitignore
| README.md
| structura_de_archivos.txt
|
+---deploy
| docker-compose.yml
|
+---docs
| alcance.md
| api-contratos.md
| arquitectura-Diagrama Contexto.drawio.png
| arquitectura-Diagrama Definici n de flujos.drawio.png
| arquitectura.drawio
| contexto.md
| modelo-datos.md
| Presentaci n - TuGranjita.pptx
| retos.md
| Tema 1 Documentaci n.pdf
|
+---schemas
| ClienteProveedor.schema.json
| ejemplo.schema.json
| Empleado.schema.json
| Ganado.schema.json
| ItemInventario.schema.json
| LecturaSensor.schema.json
| SensorIoT.schema.json
| TransaccionCompraGanado.schema.json
| TransaccionCompralInventario.schema.json
| TransaccionVentaGanado.schema.json
```

# Estructura de Archivos (TuGranjita)

---

```
\---services
    +---crm
        |   |   index.js
        |   |   main.py
        |   |   package.json
        |   |   README.md
        |   |   __init__.py
        |
        |   \---data
        |       clientes.json
        |       pedidos.json
        |
        \---iot
            |   index.js
            |   main.py
            |   package.json
            |   README.md
            |   __init__.py
            |
            \---data
                lecturas.json
                sensores.json
```

# ¿Preguntas?

Gracias por su atención.