



SIMCII



01/12/25

Instituto Tecnológico de Orizaba

Tecnologías de Programación

oooo

Dr. Ulises Juárez Martínez



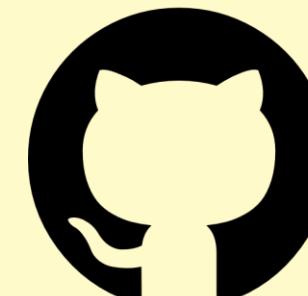
Sistema de Monitoreo y Control de Invernaderos Inteligentes

Integrantes:

Jonathan Uriel Vega Román

Emmanuel Torres Martínez

Rubén Castillo Ramírez



https://github.com/ETM95/SIMCII_

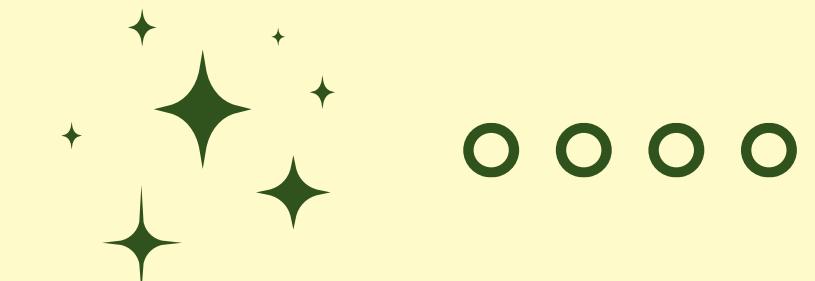
¿Que es SIMCII?

Es una plataforma de monitorización y gestión de IoT para invernaderos la cual implementa una arquitectura de microservicios servicios contenedORIZADOS que gestionan la gestión de dispositivos, la monitorización en tiempo real, las alertas y la autenticación de usuarios.

SISTEMA INVERNADERO



Componentes del sistema



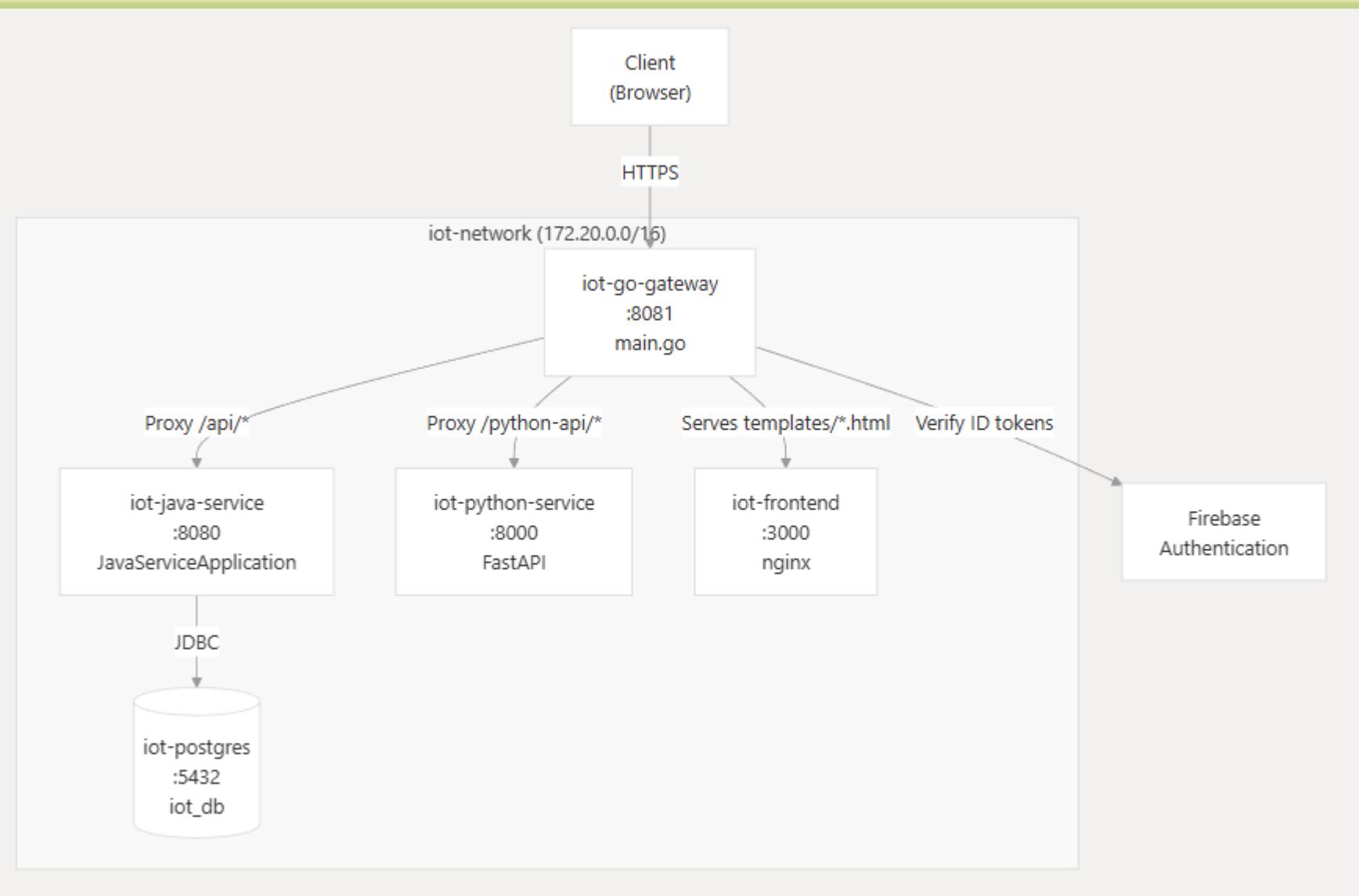
Servicio	Nombre del contenedor	Puerto	Responsabilidad primaria	Tecnología
Go Gateway	iot-go-gateway	8081	Puerta de enlace API, autenticación, enrutamiento de solicitudes, servicio de plantillas HTML	Go
Servicio Java	iot-java-service	8080	Operaciones CRUD del dispositivo, persistencia de la base de datos	Spring Boot , Java
Servicio de Python	iot-python-service	8000	Cálculo de estadísticas, gestión de alertas	RestAPI,Flask, Python
PostgreSQL	iot-postgres	5432	Persistencia de datos relacionales	PostgreSQL
Interfaz	iot-frontend	3000	Servicio de activos estáticos	Nginx Alpine



Arquitectura de alto nivel

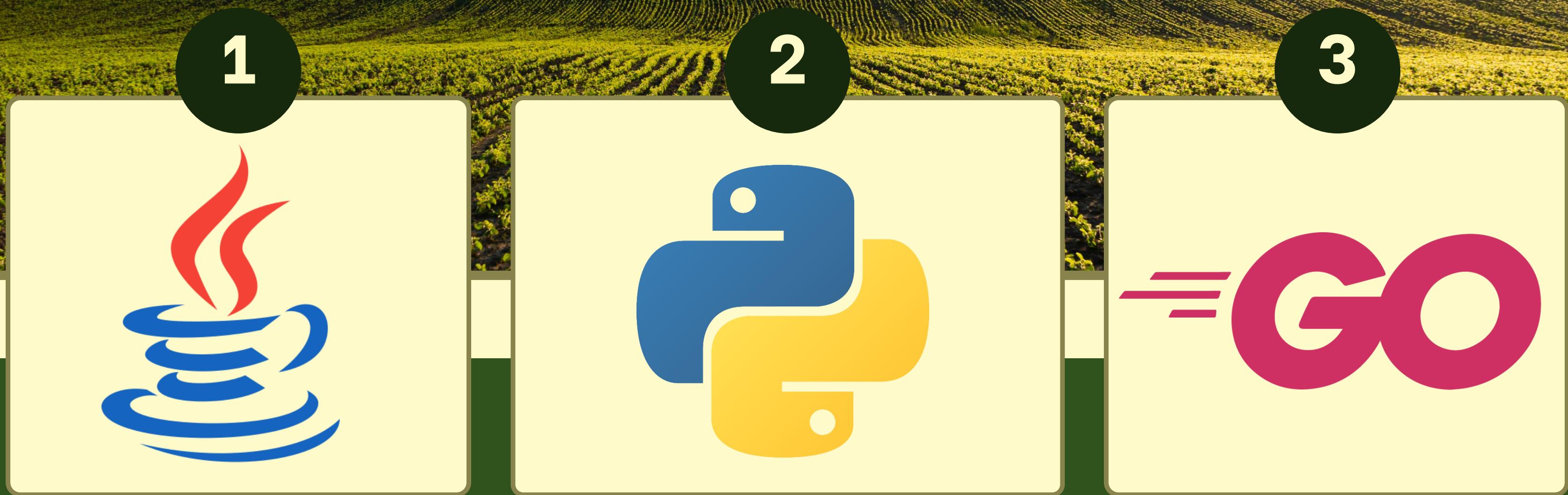


Topología del servicio

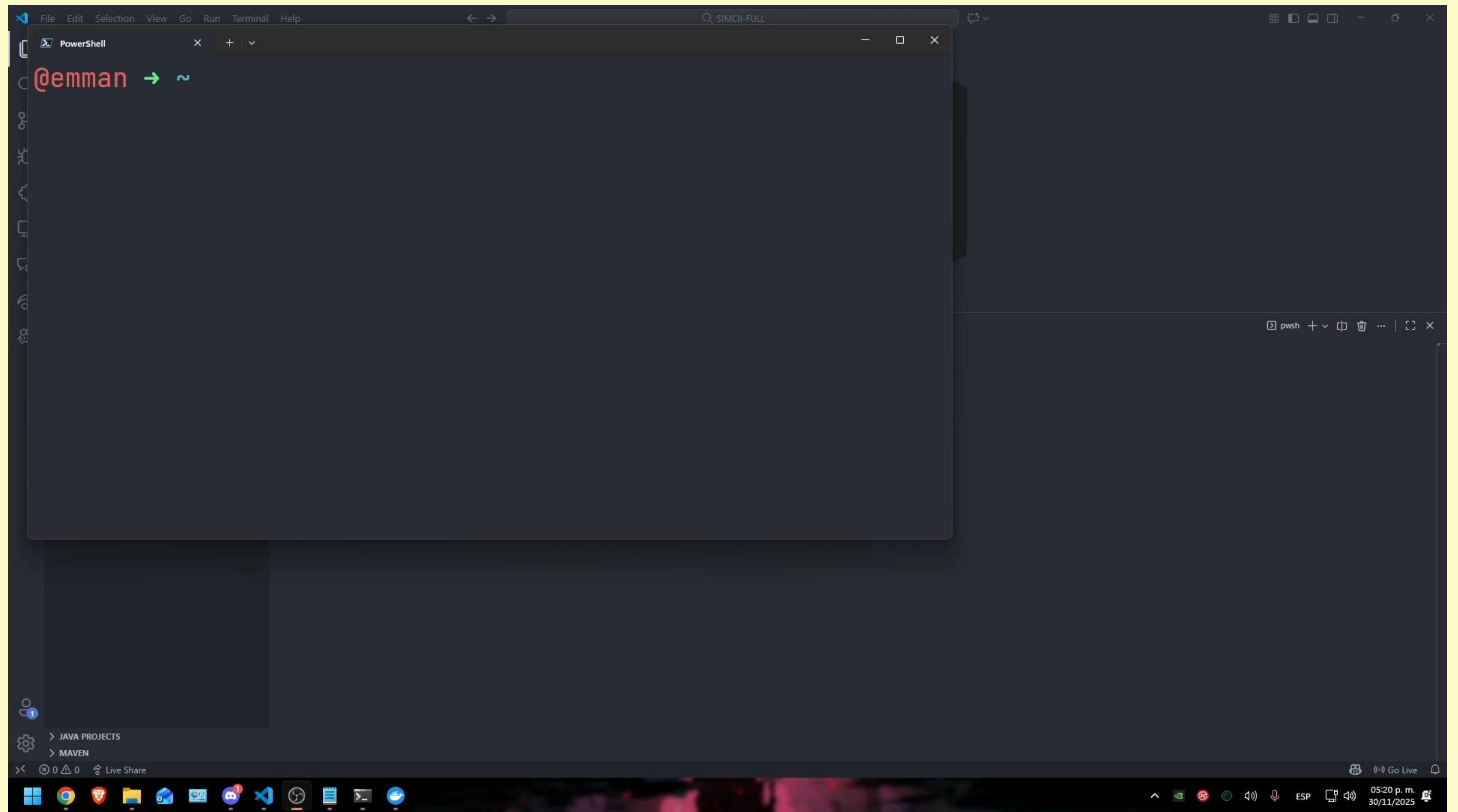


El sistema sigue un patrón de API Gateway donde todas las solicitudes de los clientes fluyen a través de la puerta de enlace Go, que maneja la autenticación y dirige el tráfico a servicios backend especializados.

Funcionamiento de cada una de los servicios



Instalación



Documentación detallada

Wiki profunda ETM95/SIMCII

Indexa tu código con  Devin

 Editar wiki

 Compartir



Última indexación: 29 de noviembre de 2025 (007869)

Descripción general

Empezando

Implementación con Docker
Compose

Configuración

Arquitectura del sistema

Comunicación de servicios

Flujo de datos

Servicio Go Gateway

Descripción general de la puerta de enlace y punto de entrada principal

Sistema de autenticación

Enrutamiento de solicitudes y middleware

Dependencias y bibliotecas

Implementación de puerta de enlace

Servicio Java

Descripción general del servicio Java

Dependencias y configuración

Construcción e implementación

Aplicación frontend

Descripción general

> Archivos fuente relevantes

Propósito y alcance

Este documento ofrece una introducción general a **SIMCII** (Sistema Invernadero), una plataforma de monitorización y gestión de IoT para invernaderos. SIMCII implementa una arquitectura de microservicios con cinco servicios contenedorizados que gestionan la gestión de dispositivos, la monitorización en tiempo real, las alertas y la autenticación de usuarios.

Esta descripción general cubre:

- Identificación y funciones de los componentes del sistema
- Arquitectura de alto nivel e interacciones de servicios
- Pila de tecnología y modelo de implementación
- Puntos de entrada clave para comprender el código base

Para obtener información detallada sobre subsistemas específicos, consulte:

- Procedimientos de implementación: [Primeros pasos](#)
- Patrones de arquitectura detallados: [Arquitectura del sistema](#)
- Documentación de servicios individuales: [Servicio Go Gateway](#) , [Servicio Java](#) , [Aplicación Frontend](#)
- Implementación de seguridad: [Autenticación y seguridad](#)

Pregúntale a Devin sobre ETM95/SIMCII

← Rápido ↗

En esta página

Descripción general

Propósito y alcance

Componentes del sistema

Arquitectura de alto nivel

Topología del servicio

Flujo de solicitud y autenticación

Ciclo de vida típico de una solicitud autenticada

Pila de tecnología por capa

Capa de puerta de enlace

Capa de servicio

Capa de datos

Capa frontal

Dependencias de servicio y comprobaciones de estado

Puntos de entrada clave

Punto de entrada de la puerta de enlace

Punto de entrada del servicio Java

Configuración de red

Configuración del entorno

Variables de entorno de puerta de enlace

Variables de entorno del servicio Java

Gracias por su
atención

¿Preguntas?