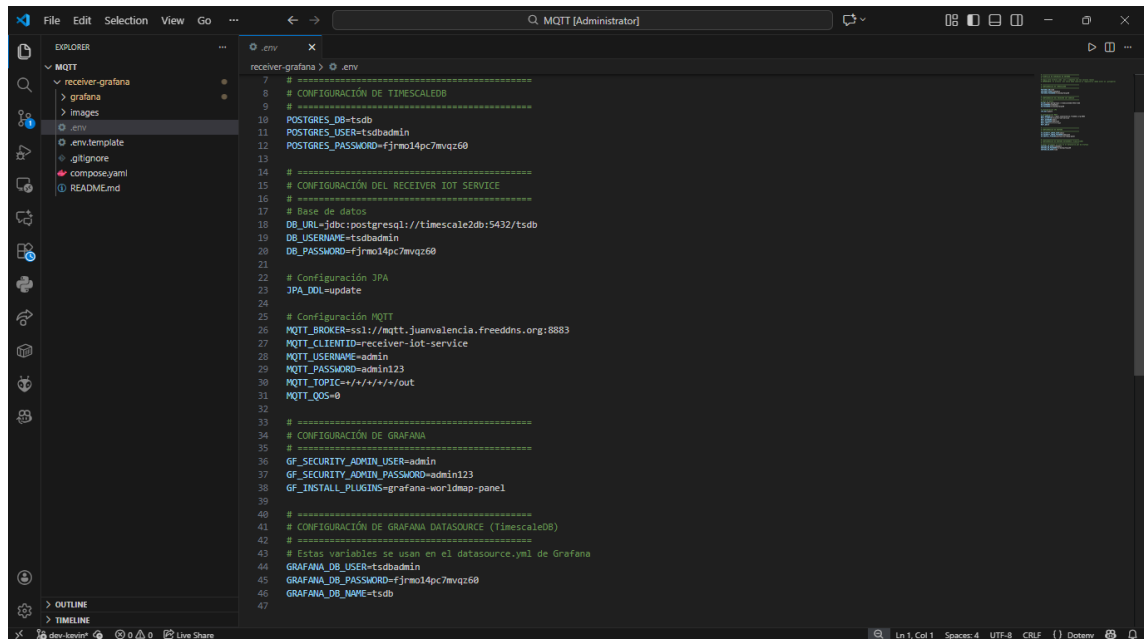


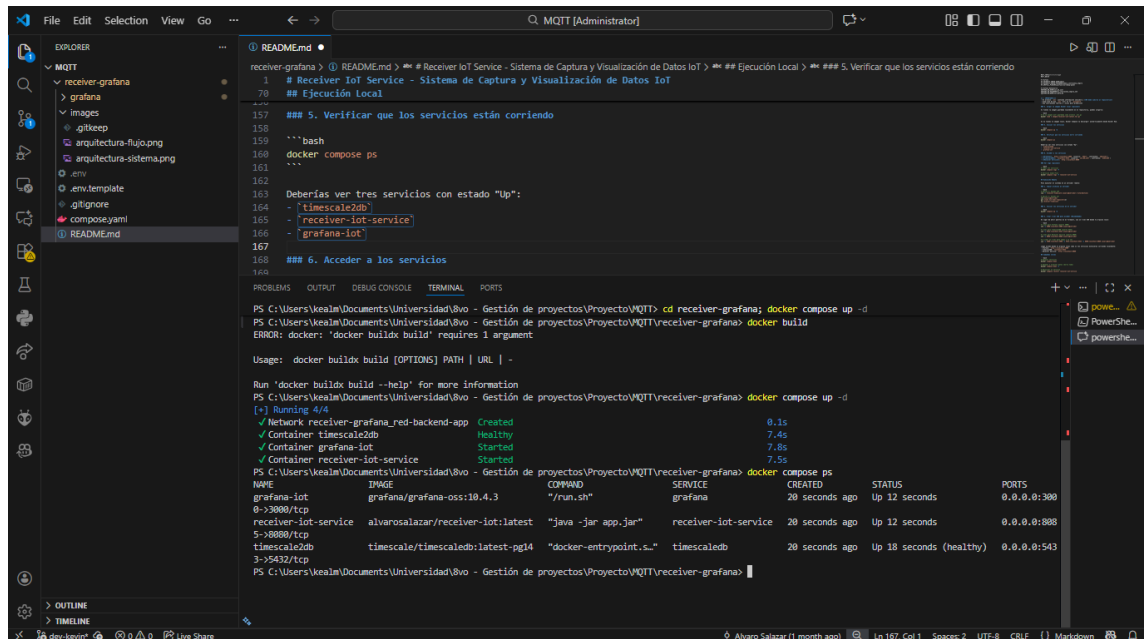
Documentación Codelab IV

1. Primero se configura el .env



```
receiver-grafana > .env
7 # =====
8 # CONFIGURACIÓN DE TIMESCALEDB
9 # =====
10 POSTGRES_DB=tsdb
11 POSTGRES_USER=tsdbadmin
12 POSTGRES_PASSWORD=fjrm014pc7mqz60
13
14 # =====
15 # CONFIGURACIÓN DEL RECEIVER IOT SERVICE
16 # =====
17 # Base de datos
18 DB_URL=jdbc:postgresql://timescale2db:5432/tsdb
19 DB_USERNAME=tsdbadmin
20 DB_PASSWORD=fjrm014pc7mqz60
21
22 # Configuración JPA
23 JPA_DDL-update
24
25 # Configuración MQTT
26 MQTT_BROKER=ssl://mqtt.juanvalencia.freedomns.org:8883
27 MQTT_CLIENTID=receiver-iot-service
28 MQTT_USERNAME=admin
29 MQTT_PASSWORD=admin123
30 MQTT_TOPIC=+/+/+/+/out
31 MQTT_QOS=0
32
33 # =====
34 # CONFIGURACIÓN DE GRAFANA
35 # =====
36 GF_SECURITY_ADMIN_USER=admin
37 GF_SECURITY_ADMIN_PASSWORD=admin123
38 GF_INSTALL_PLUGINS=grafana-worldmap-panel
39
40 # =====
41 # CONFIGURACIÓN DE GRAFANA DATASOURCE (timescaledb)
42 # =====
43 # Estas variables se usan en el datasource.yml de Grafana
44 GRAFANA_DB_USER=tsdbadmin
45 GRAFANA_DB_PASSWORD=fjrm014pc7mqz60
46 GRAFANA_DB_NAME=tsdb
47
```

2. Luego abrimos Docker desktop y cargamos la imagen con “docker compose up -d” y luego ponemos “docker compose ps” para verificar que se estén cargando correctamente.



```
receiver-grafana > README.md
1 # Receiver IoT Service - Sistema de Captura y Visualización de Datos IoT
2 # Ejecución Local
3
4 ## Ejecución Local
5
6 ### 5. Verificar que los servicios están corriendo
7
8 ```bash
9 docker compose ps
10 ```
11
12 Deberías ver tres servicios con estado "Up":
13 - timescale2db
14 - receiver-iot-service
15 - grafana-iot
16
17 ### 6. Acceder a los servicios
18
```

```
PS C:\Users\kealm\Documents\Universidad\8vo - Gestión de proyectos\Proyecto\MQTT\receiver-grafana> cd receiver-grafana; docker compose up -d
PS C:\Users\kealm\Documents\Universidad\8vo - Gestión de proyectos\Proyecto\MQTT\receiver-grafana> docker build
ERROR: docker: 'docker build' requires 1 argument
Usage: docker build build [OPTIONS] PATH | URL | -
Run 'docker build --help' for more information
PS C:\Users\kealm\Documents\Universidad\8vo - Gestión de proyectos\Proyecto\MQTT\receiver-grafana> docker compose up -d
[+] Running 4/4
 ✓ Network receiver-grafana_red-backend-app Created 0.1s
 ✓ Container timescale2db Healthy 7.4s
 ✓ Container grafana-iot Started 7.8s
 ✓ Container receiver-iot-service Started 7.5s
PS C:\Users\kealm\Documents\Universidad\8vo - Gestión de proyectos\Proyecto\MQTT\receiver-grafana> docker compose ps
NAME                IMAGE                                COMMAND                SERVICE        CREATED        STATUS        PORTS
grafana-iot          grafana/grafana-oss:10.4.3          "/run.sh"             grafana        20 seconds ago Up 12 seconds 0.0.0.0:300
receiver-iot-service alvarosalazar/receiver-iot:latest   "java -jar app.jar"    receiver-iot-service 20 seconds ago Up 12 seconds 0.0.0.0:808
timescale2db         timescale/timescaledb:latest-pg14   "docker-entrypoint.s..." timescaledb     20 seconds ago Up 18 seconds (healthy) 0.0.0.0:543
```

3. Si funciona bien, abrimos el localhost:3000 y deberíamos ver la interfaz de grafana y accedemos con las credenciales que configuramos en el archivo “.env”

