

**INSTITUTO DE EDUCACIÓN SUPERIOR CIBERTEC**

NOTA

**DIRECCIÓN ACADÉMICA CARRERA PROFESIONALES**

**CURSO :** 0268 – Desarrollo de Aplicaciones Web II

**SEMESTRE :** 2024-5

**CICLO :** Sexto

**ALUMNO (A) : WILMER OCAMPO QUISPE**

EXAMEN FINAL

**Logro:**

Tenga presente que el logro de esta evaluación es que usted implemente la creación de un sistema de seguridad que incluye la creación de usuarios, login y acceso a otros servicios.

**Consideraciones generales:**

* El desarrollo de la práctica es estrictamente individual.
* Las preguntas deben responderse de acuerdo con los conceptos discutidos o analizados en clase y aplicando su propio criterio para dar una solución a los problemas planteados.

**Consolidado**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pregunta** | **Puntaje** | | **Llenar solo en caso de Recalificación justificada** | |
| **Máximo** | **Obtenido** | **Sustento** | **Puntaje** |
| **1** | **20** |  |  |  |
| **Nota Recalificada** | | | |  |

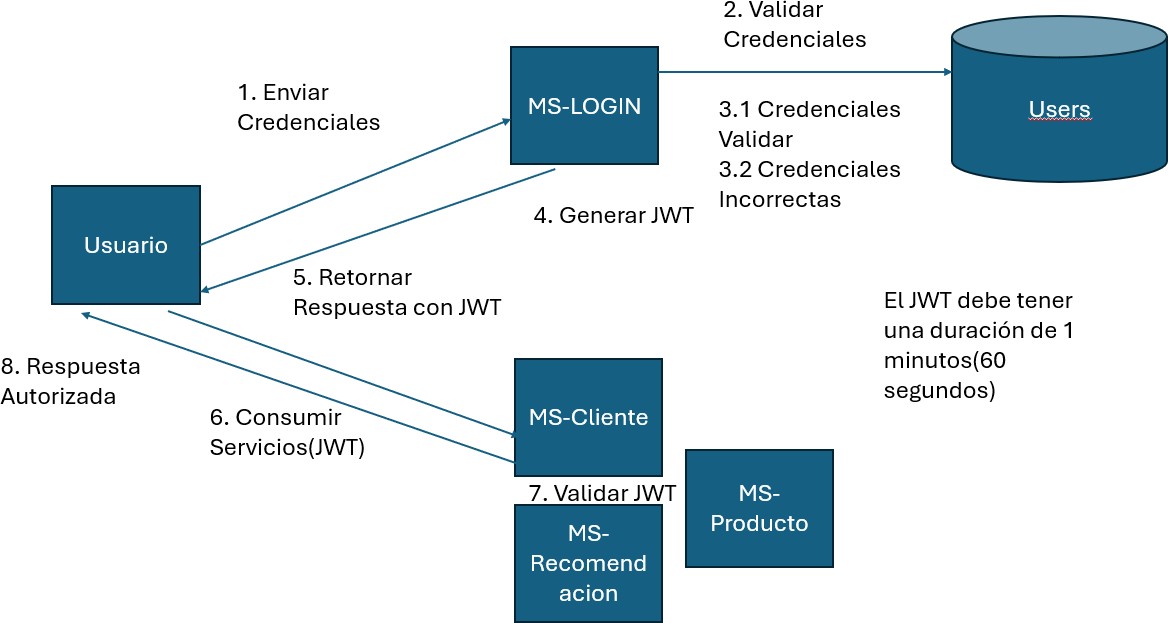
**LINK DEL REPOSITORIO GIT HUB:** [**WilmerOcampo/DAWII\_EF\_OcampoWilmer**](https://github.com/WilmerOcampo/DAWII_EF_OcampoWilmer)

[**https://github.com/WilmerOcampo/DAWII\_EF\_OcampoWilmer**](https://github.com/WilmerOcampo/DAWII_EF_OcampoWilmer)

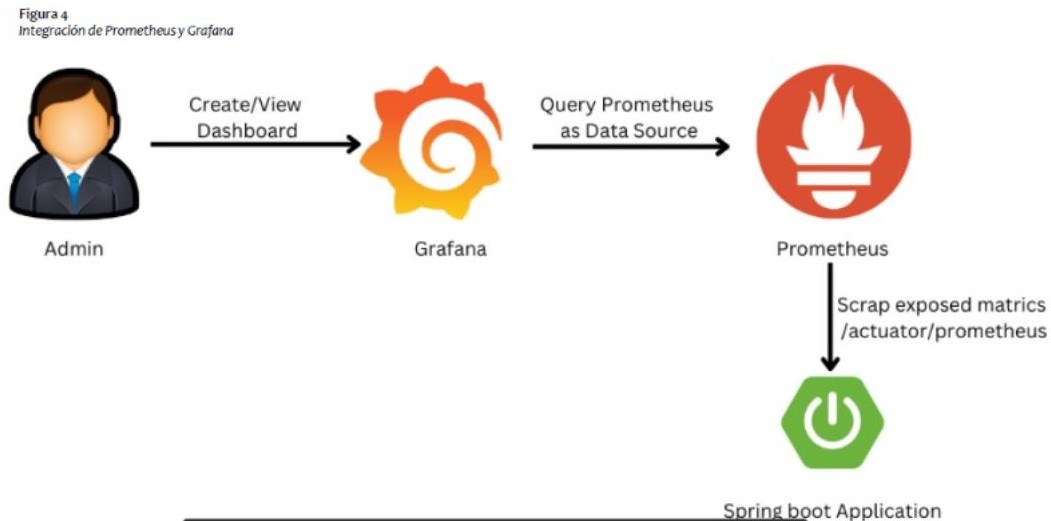
**Pregunta 01**

Usted como desarrollador se le asignado la funcionalidad de construir 3 servicios de clientes, productos y recomendaciones desarrollado con spring boot.

Asimismo, se le solicita implementar la seguridad en los servicios que 1 servicios adicional que gestione la creación de usuarios(usuario,password), login y generación de JWT con expiración de 1 minuto(60 segundos) con spring security en el motor de base de datos MYSQL.



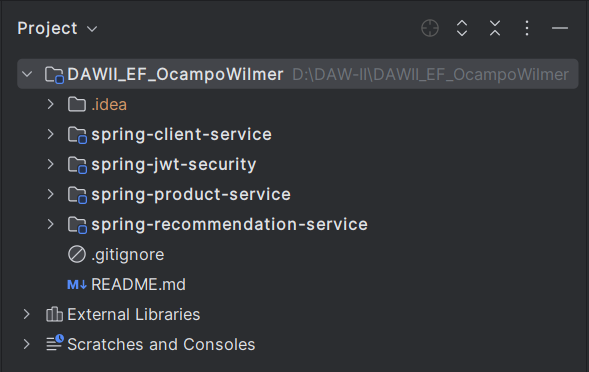
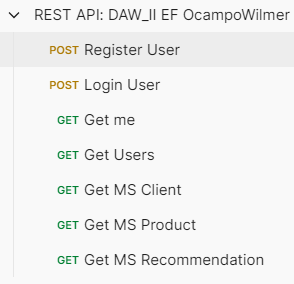
Además, se solicita habilitar el monitoreo de todos los servicios, incluyendo seguridad con spring boot actuator, prometheus y grafana. Se debería poder acceder a la consola de Grafana y ver el estado de los 4 microservicios.



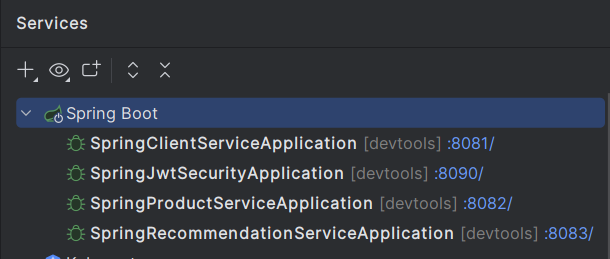
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RÚBRICA DE LA EVALUACIÓN** | | | | |
| **Puntaje** | **Criterios de Evaluación** | | | |
| **Funcionalidad de microservicios (20 Puntos)** | **Excelente: 20puntos** | **Bueno:**  **15 puntos** | **Regular:**  **7 puntos** | **Deficiente:**  **0 puntos** |
| Implementa Monitoreo de todos los servicios mencionados. Adjunta las pruebas | Implementa 1 Servicio que gestión la seguridad.  Adjunta las pruebas | Implementa los 3 microservicios producto, cliente, recomendación.  Adjunta Pruebas | No implementa Ninguna funcionalidad. |
| **Total 20 Puntos** | PUNTAJE TOTAL DEL ALUMNO | | | |

**Solucion**

**ESTRUCTURA DEL ROYECTO**

****

**ENDPOINTS POSTMAN**

****

**dependencias**

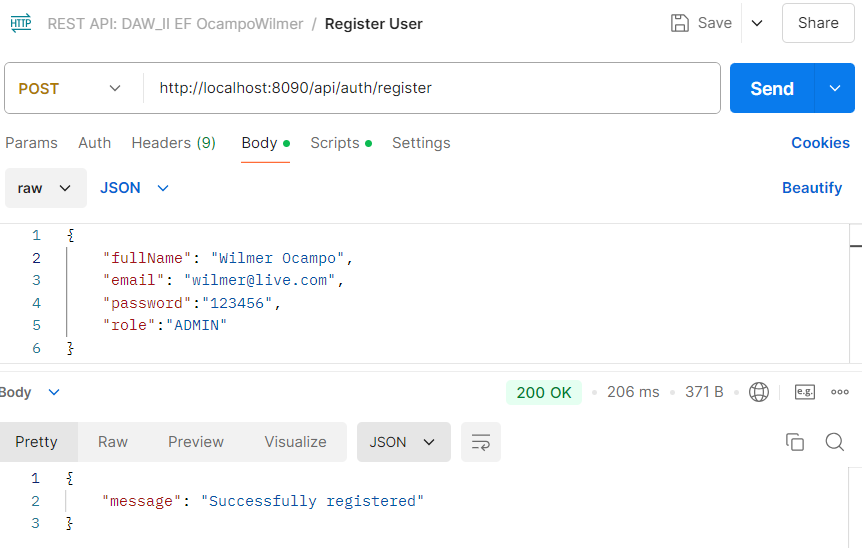
**Para JWT Token**

<!-- https://mvnrepository.com/artifact/io.jsonwebtoken/jjwt-api -->  
<dependency>  
 <groupId>io.jsonwebtoken</groupId>  
 <artifactId>jjwt-api</artifactId>  
 <version>0.12.6</version>  
</dependency>  
<!-- https://mvnrepository.com/artifact/io.jsonwebtoken/jjwt-impl -->  
<dependency>  
 <groupId>io.jsonwebtoken</groupId>  
 <artifactId>jjwt-impl</artifactId>  
 <version>0.12.6</version>  
 <scope>runtime</scope>  
</dependency>  
<!-- https://mvnrepository.com/artifact/io.jsonwebtoken/jjwt-jackson -->  
<dependency>  
 <groupId>io.jsonwebtoken</groupId>  
 <artifactId>jjwt-jackson</artifactId>  
 <version>0.12.6</version>  
 <scope>runtime</scope>  
</dependency>

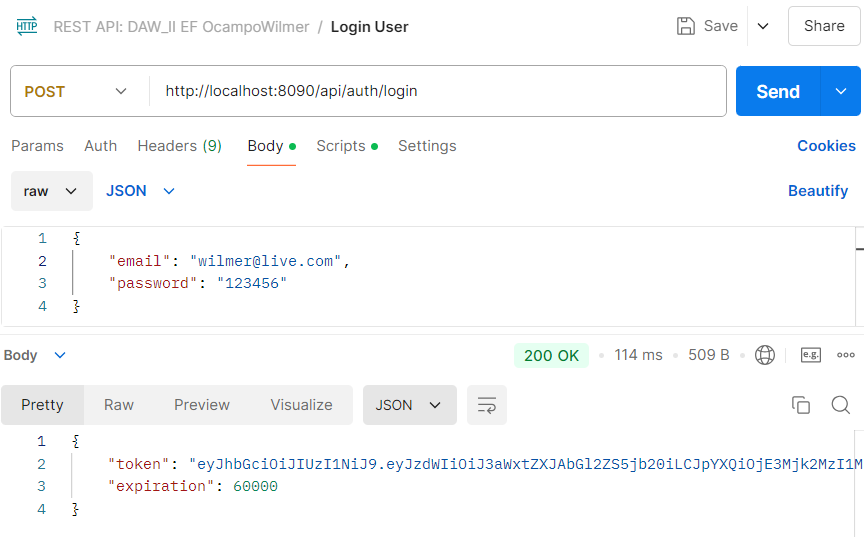
**Para Prometheus y spring boot actuator**

<dependency>  
 <groupId>io.micrometer</groupId>  
 <artifactId>micrometer-registry-prometheus</artifactId>  
 <scope>runtime</scope>  
</dependency>  
<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-actuator</artifactId>  
</dependency>

**REGISTRO USUARIO**

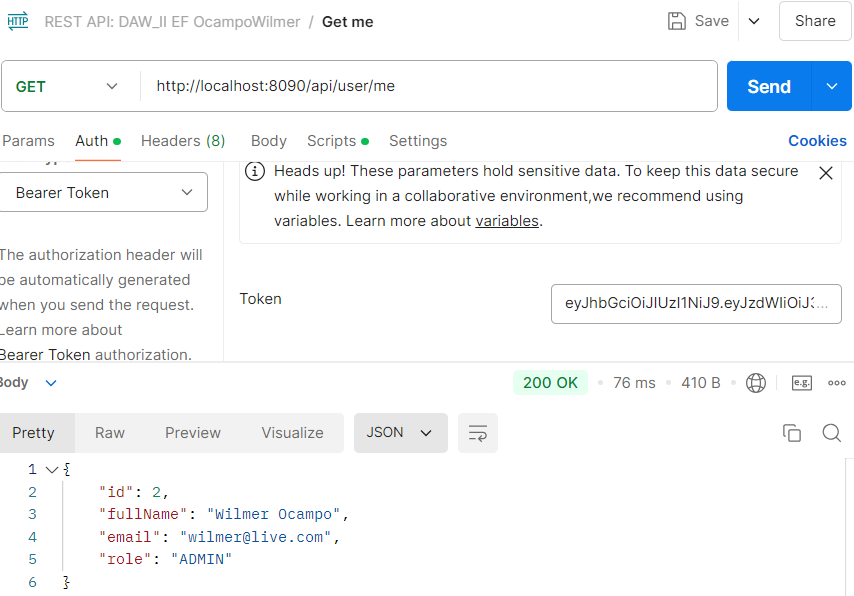
****

**LOGIN USUARIO**

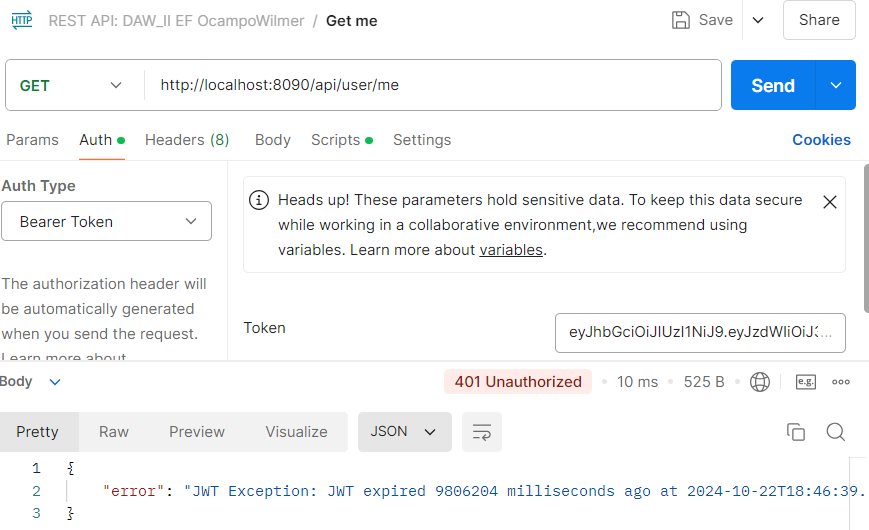
****

**ENDPOINT datos USUARIO LOGUEADO**

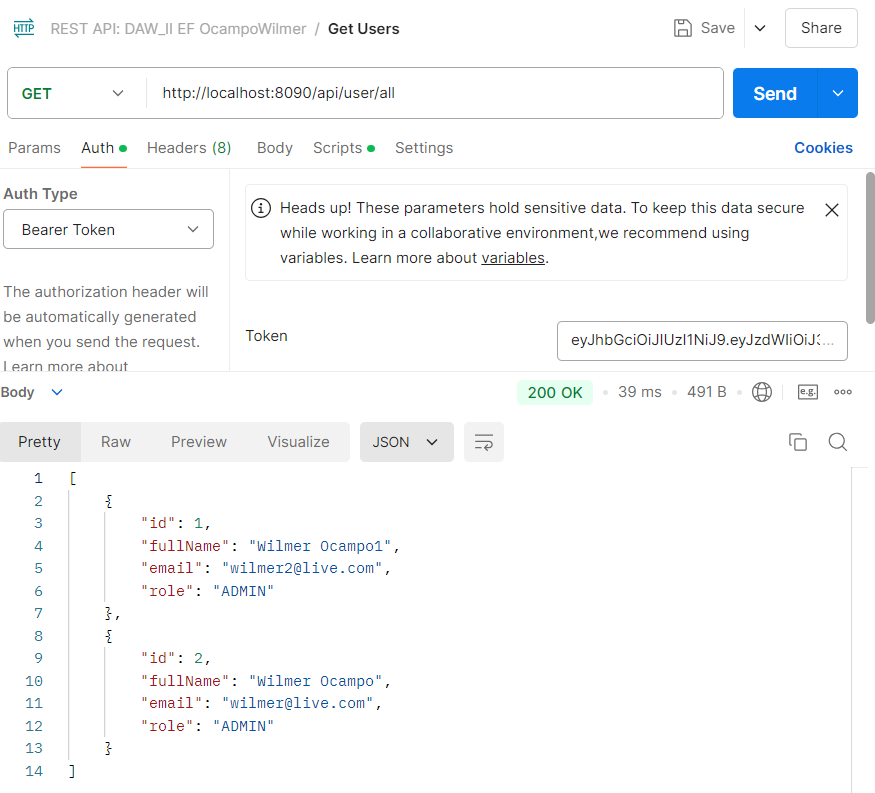
**CON TOKEN VALIDO**

****

**FORMATO CON TOKEN INVALIDO / EXPIRADO**

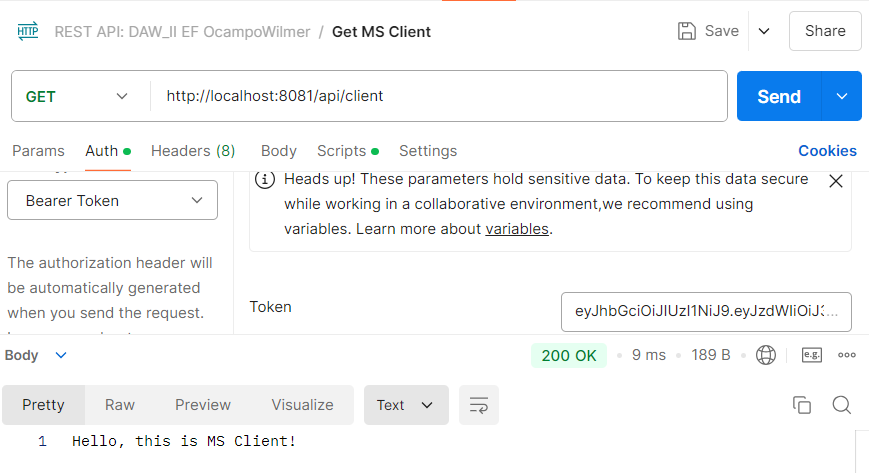
****

**ENDPOINT USUARIOS**

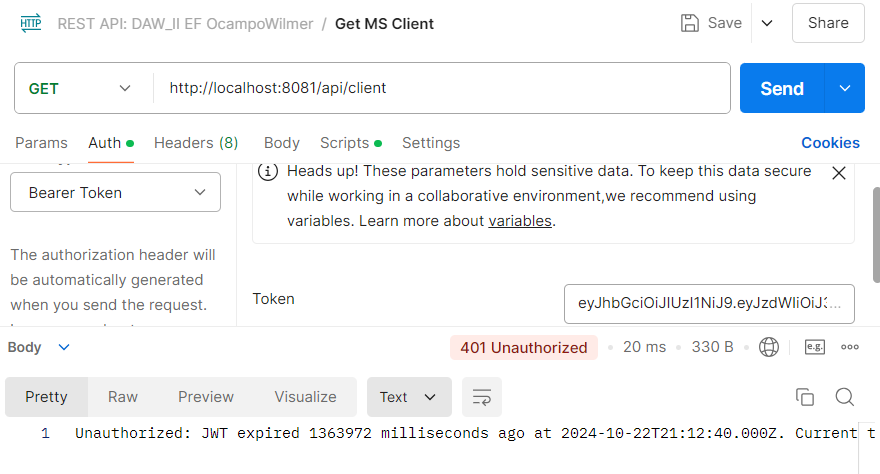
****

**ENDPOINT MICROSERVICIO CLIENT**

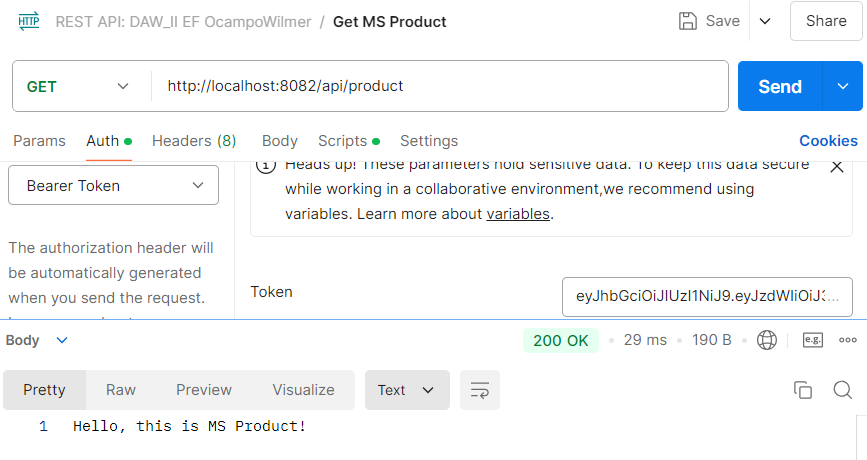
**CON TOKEN VALIDO**

****

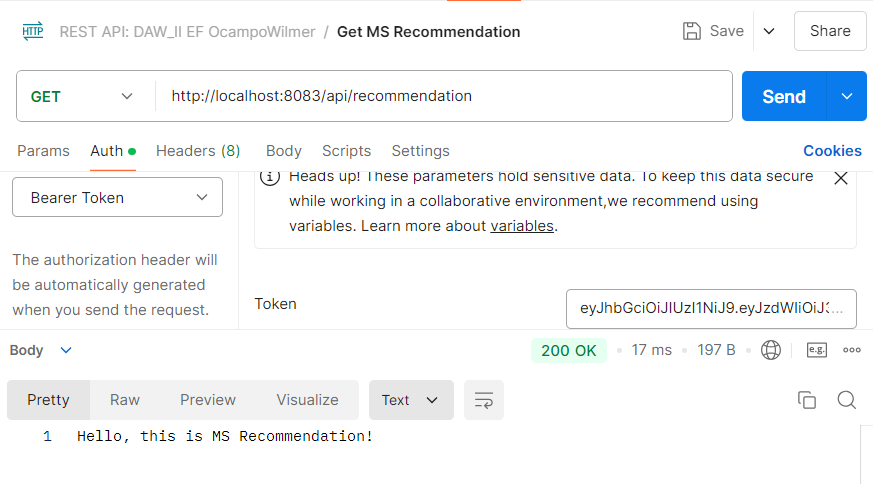
**FORMATO CON TOKEN INVALIDO / EXPIRADO**

****

**ENDPOINT MICROSERVICIO PRODUCT**

****

**ENDPOINT MICROSERVICIO RECOMMENDATION**

****

**CONFIGURACION** *prometheus.yml*

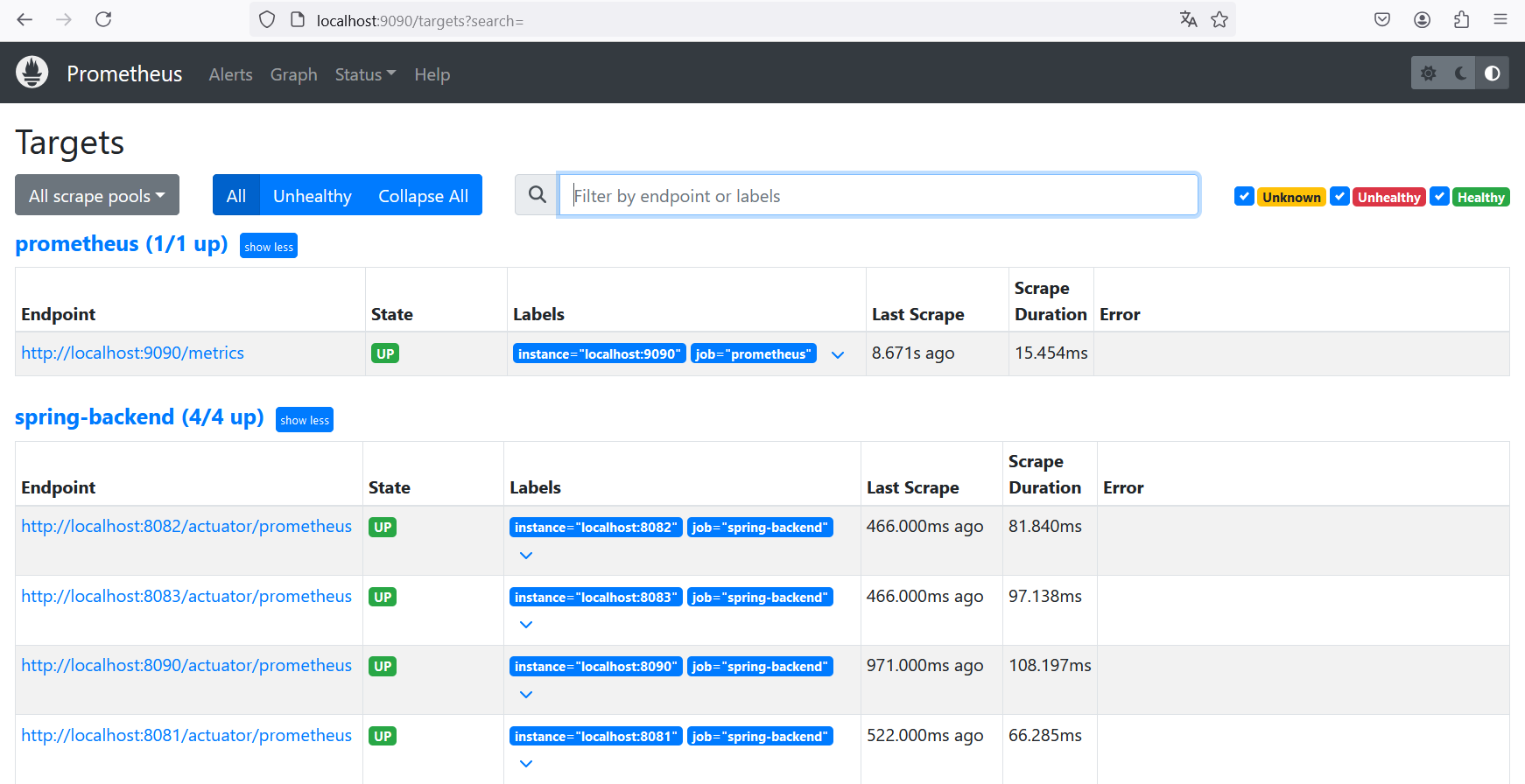
Configuracion de actuator para prometheus en los microservcios properties.yml

*# Actuator*management:  
 endpoints:  
 web:  
 exposure:  
 include: '\*'  
  
 endpoint:  
 health:  
 show-details: *always* probes:  
 enabled: true

*# Custom metrics*- job\_name: "spring-backend"  
 metrics\_path: '/actuator/prometheus'  
 scrape\_interval: 1s  
 static\_configs:  
 - targets:  
 - "localhost:8090"  
 - "localhost:8081"  
 - "localhost:8082"  
 - "localhost:8083"

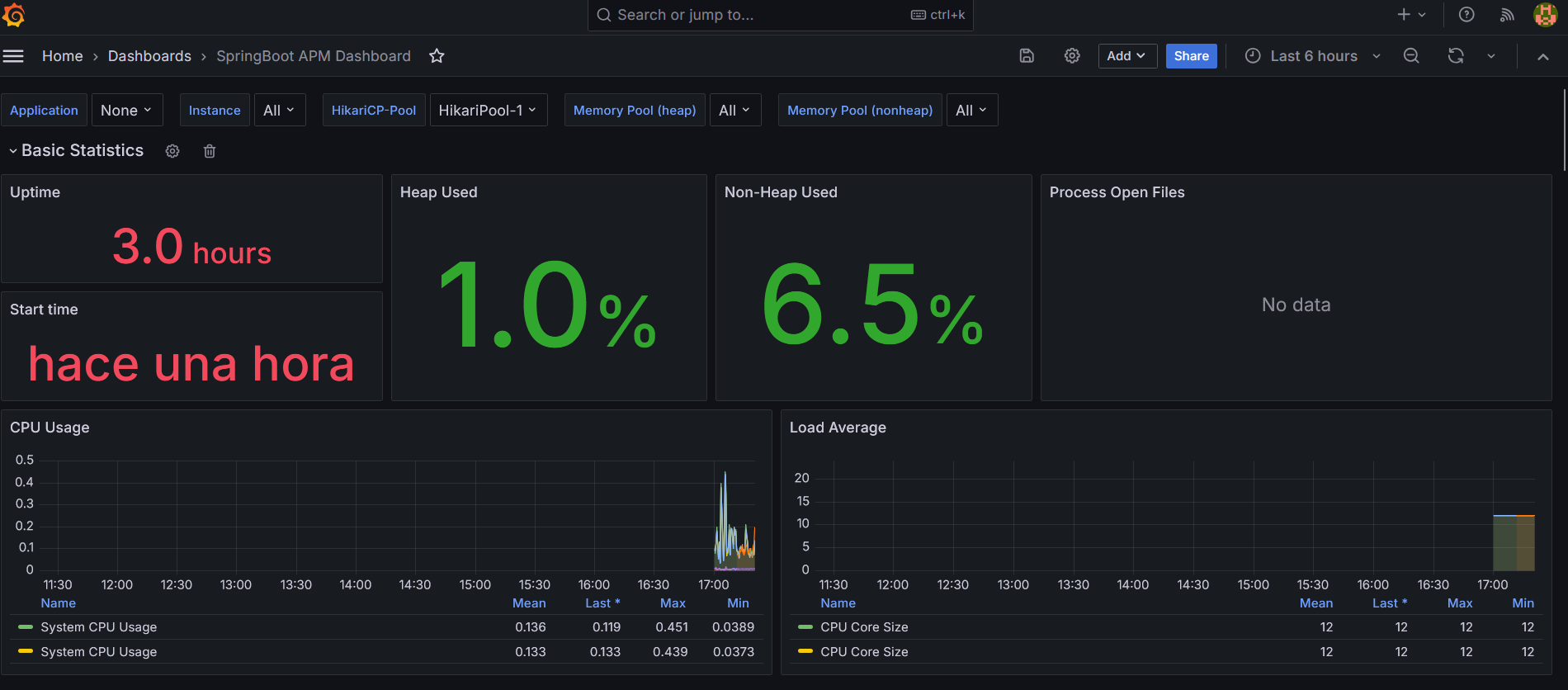
**prometheus: microservicios afiliados**

Cabe resaltar que para que pueda acceder al microservicio de jwt hace falta habilitar la ruta del actuator en ***springconfig.java***

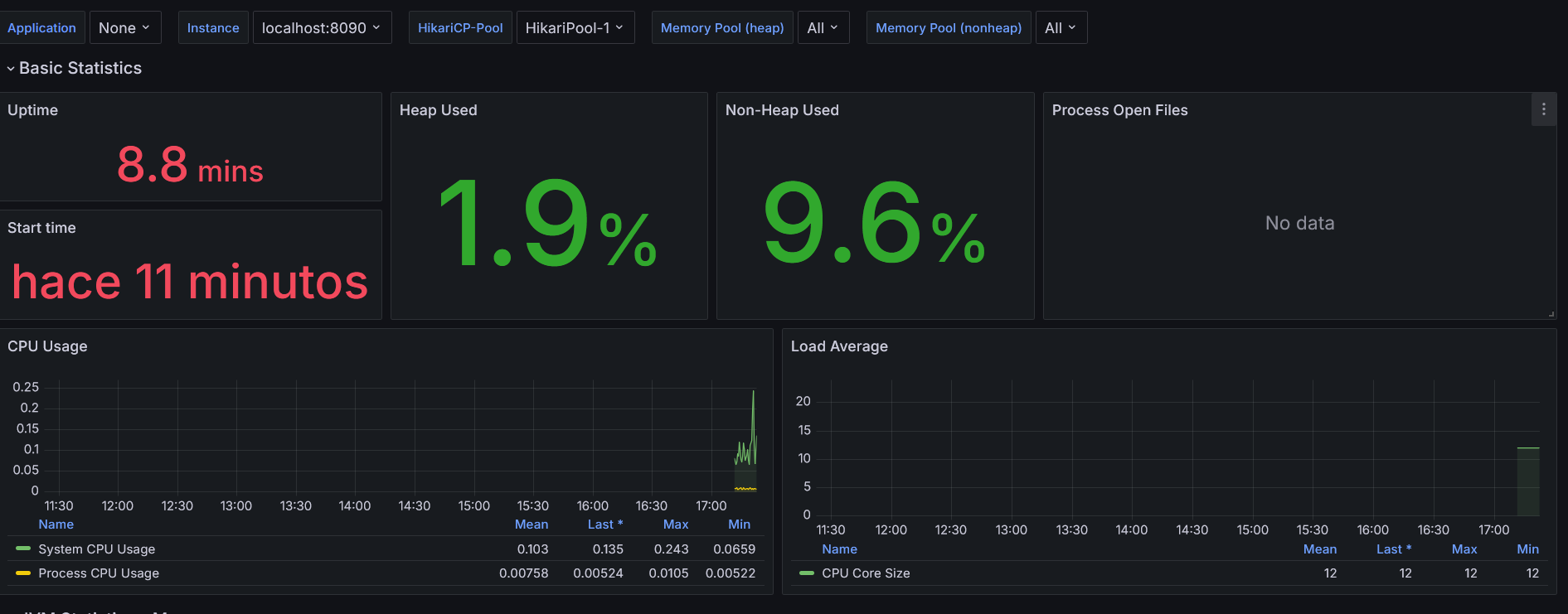
****

**grafana**

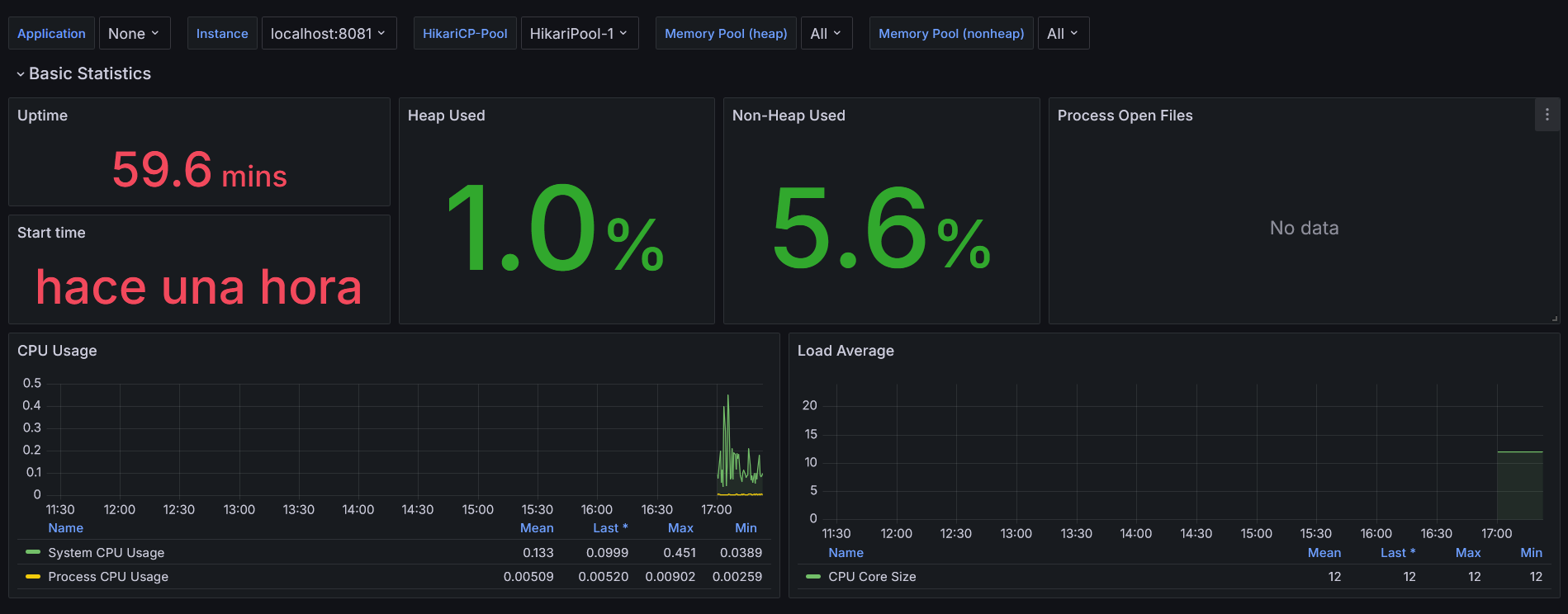
Todos los microservicios

****

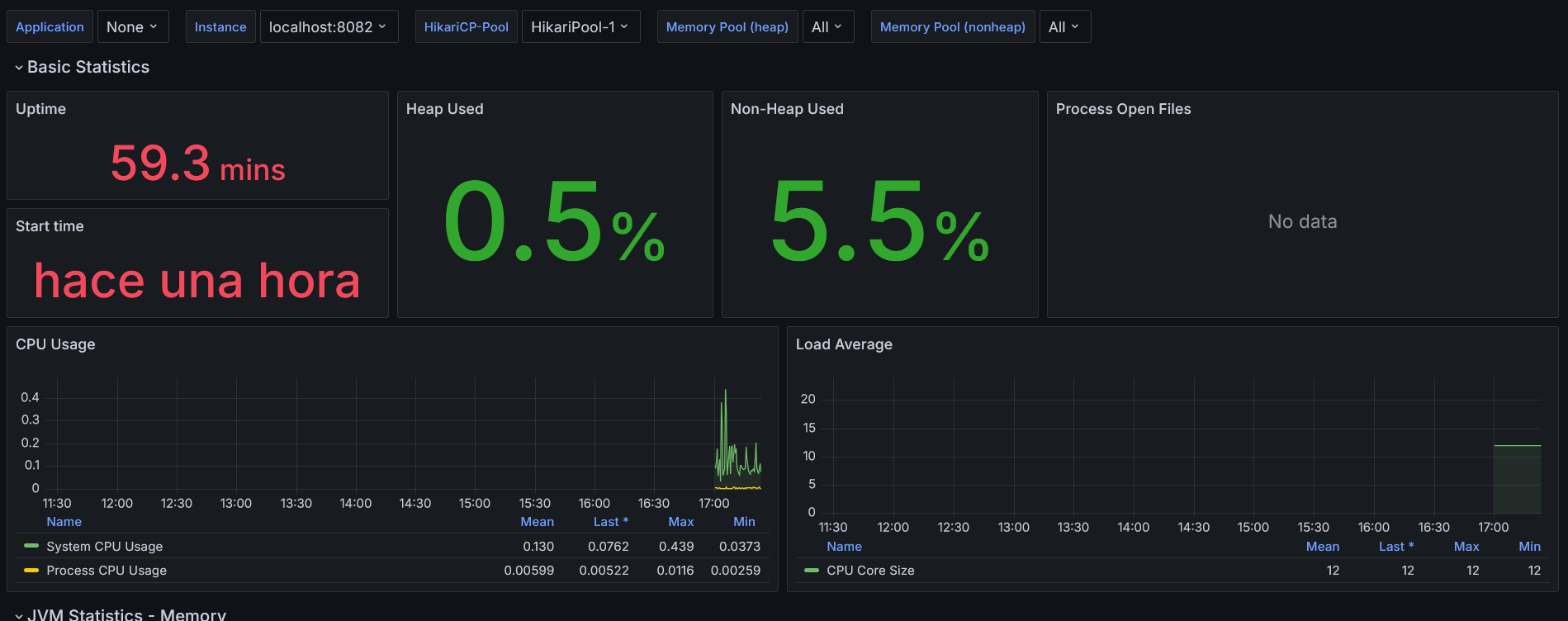
**Microservicio jwt**

****

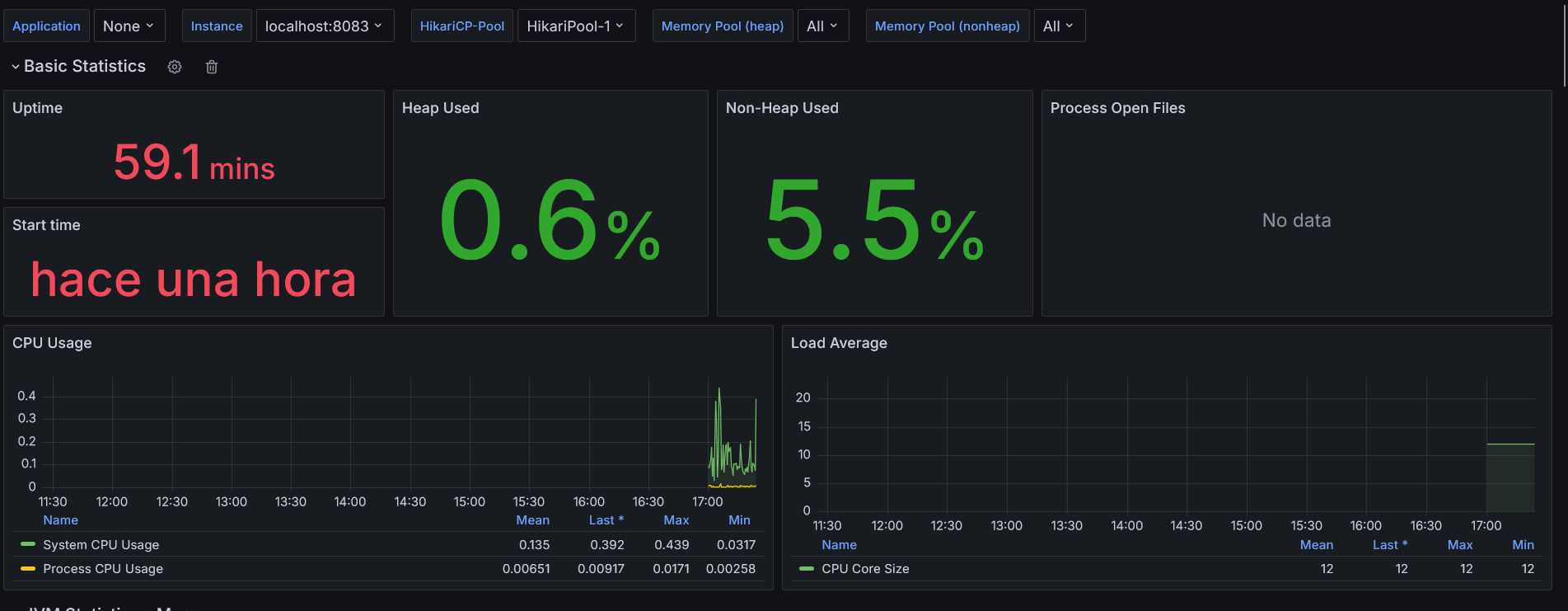
**microservicio client**



**microservicio product**

****

**microservicio recommendation**

****