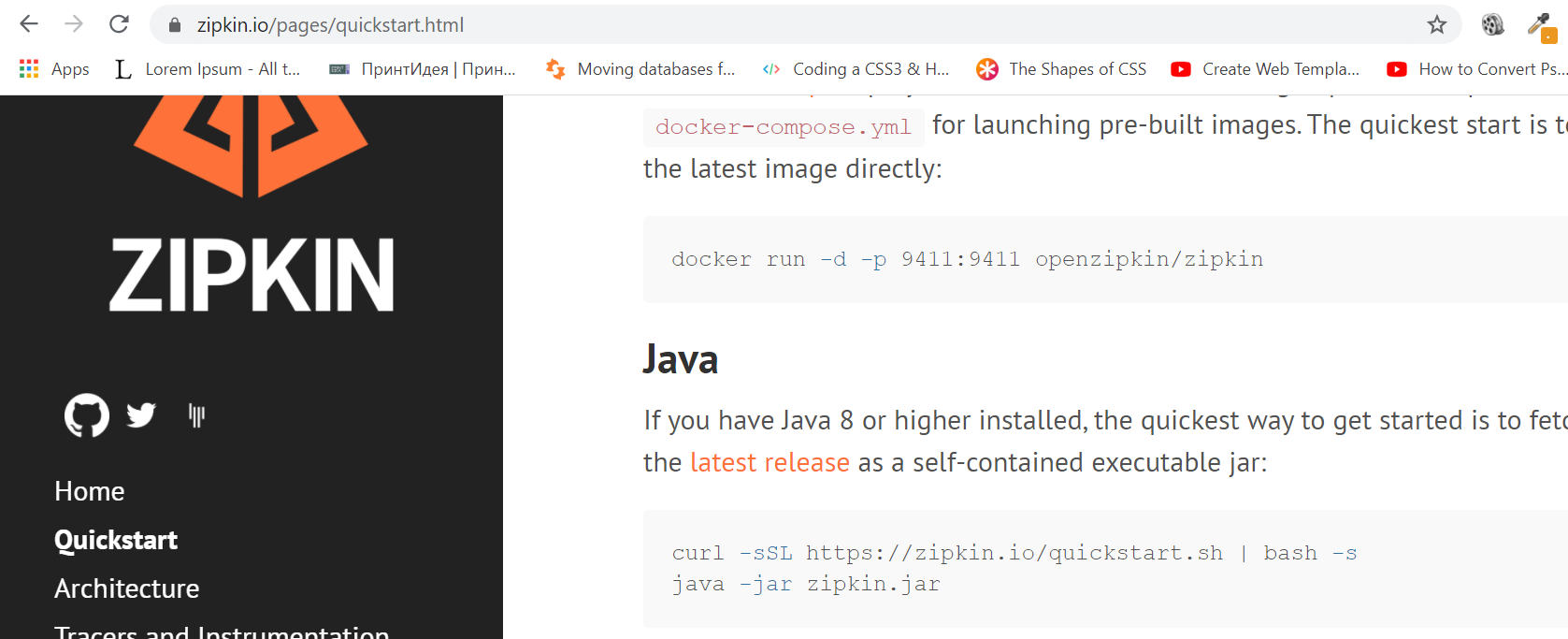
**Zipkin and Sleuth**

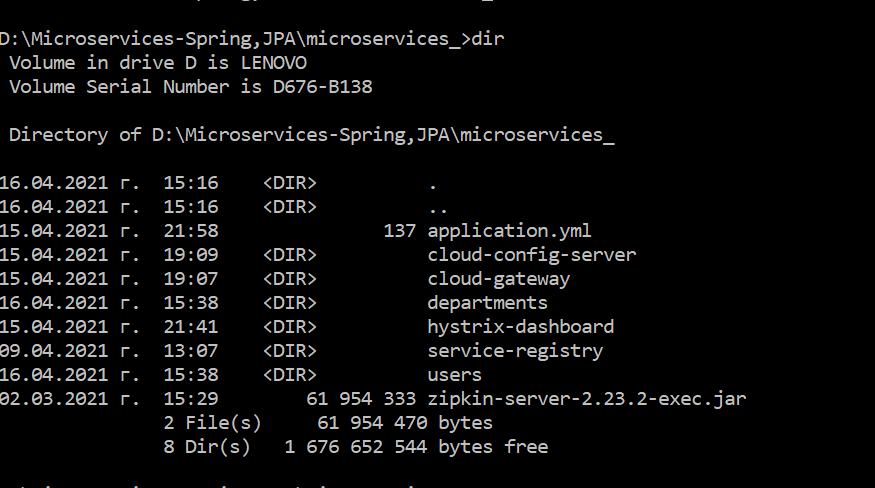
Zipkin is a distributed tracing system. It helps gather timing data needed to troubleshoot latency problems in service architectures.(https://zipkin.io)

Spring Cloud Sleuth provides Spring Boot auto-configuration for distributed tracing. Adds trace and span ids to the Slf4J MDC, so you can extract all the logs from a given trace or span in a log aggregator.( https://spring.io/projects/spring-cloud-sleuth)



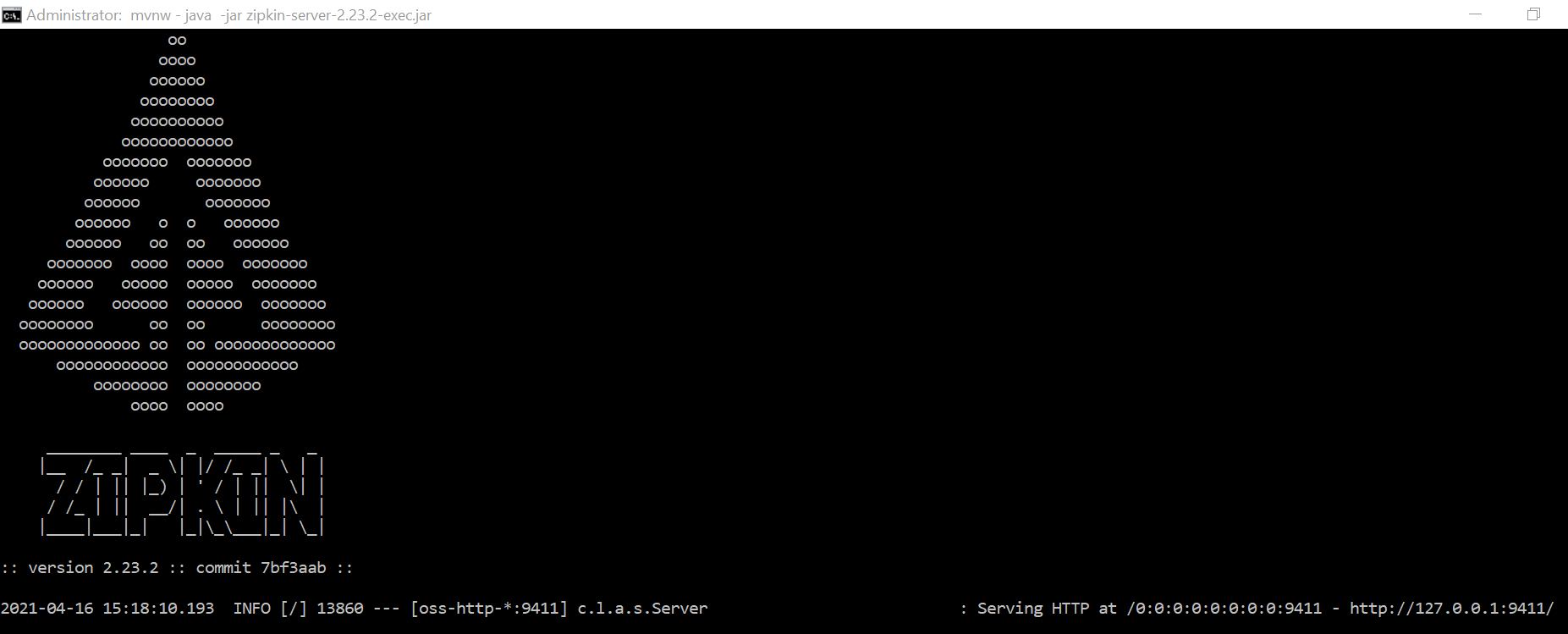
Click on Latest release to download a zipkin-server jar file.

Put this file into your project directory and perform this command from Command prompt

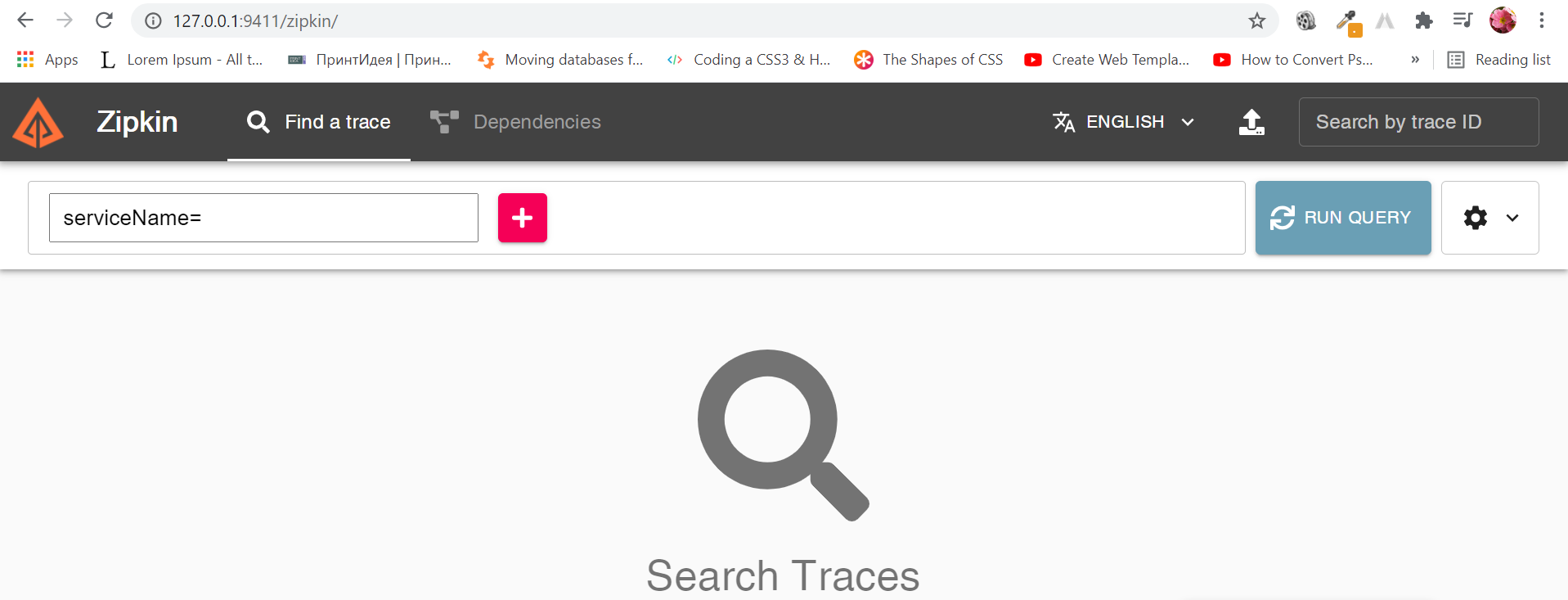


to start zipkin server:

D:\Microservices-Spring,JPA\microservices\_>java –jar zipkin-server-2.23.2-exec.jar



Copy <http://127.0.0.1:9411/> and place it into browser:



To see the names of our microservices we need to add zipkin client and sleuth dependencies to pom.xml of our services: departments and users. Go to Spring Ititializr and copy dependencies:

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-sleuth</artifactId>

</dependency>

<dependency>

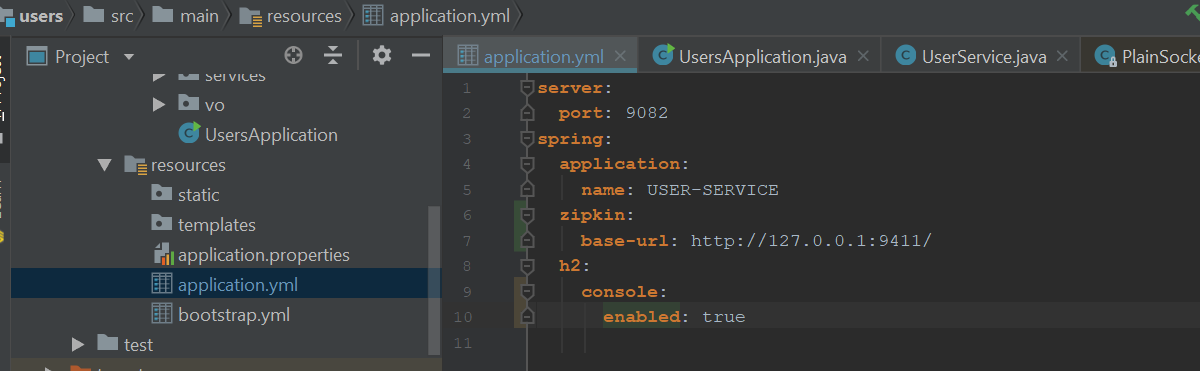
<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-zipkin</artifactId>

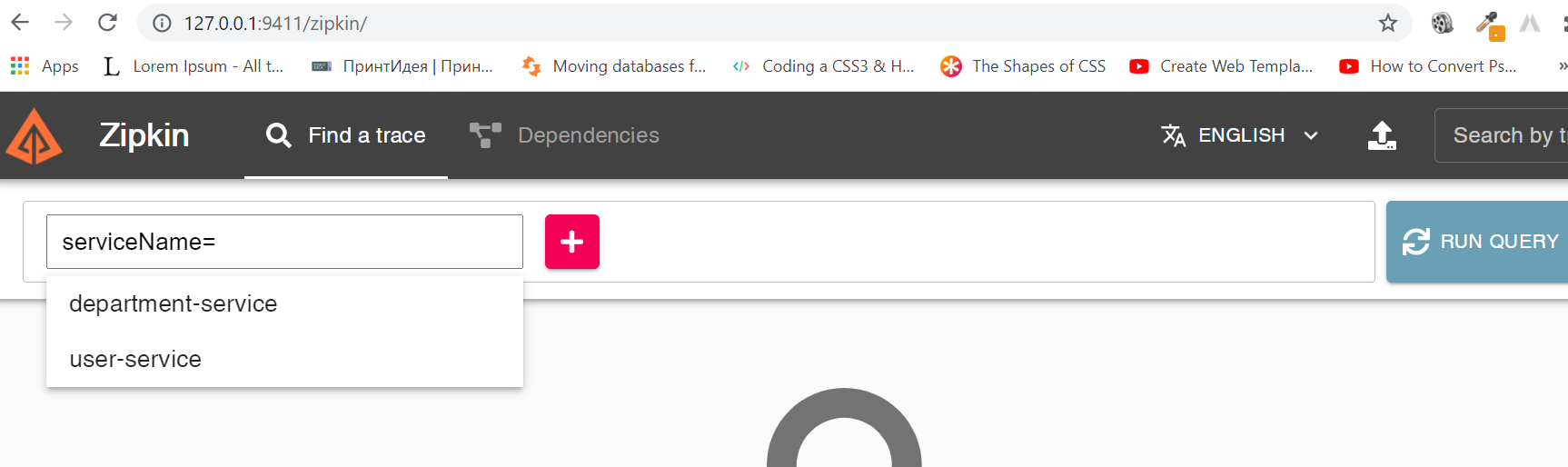
</dependency>

Then add to application.yml of both servers the code below:

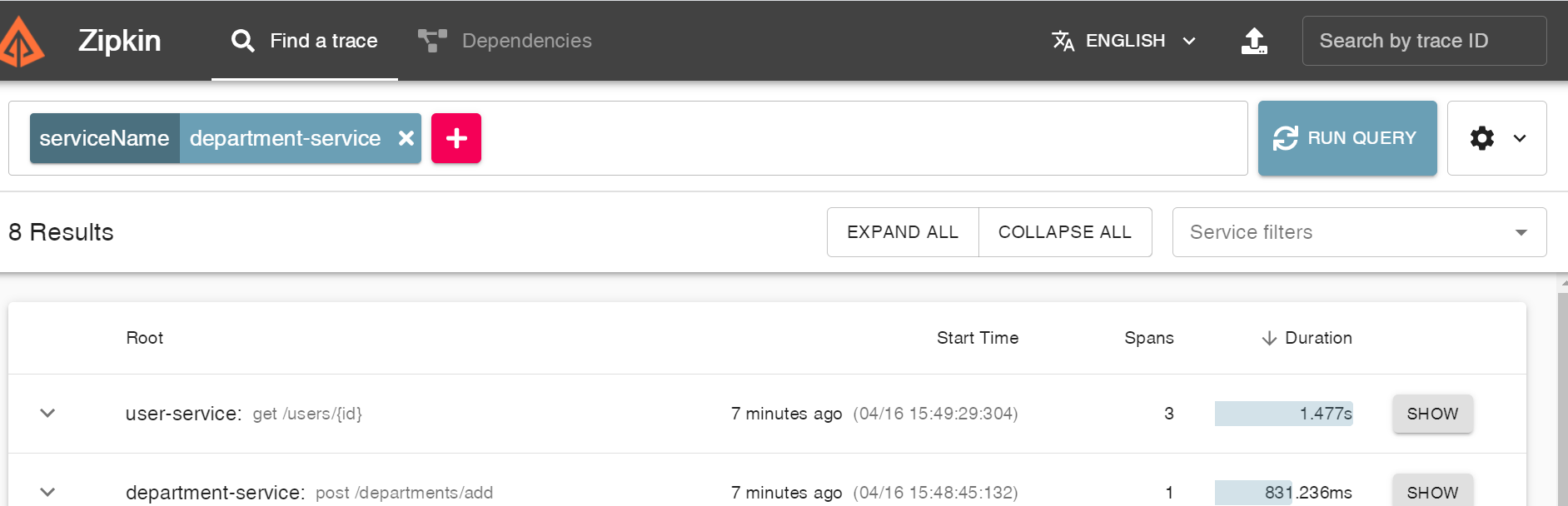
spring: zipkin: base-url: http://127:00.1:9411/



Start all servers, test requests on Postman and refresh a zipkin server:



Choose department-service and click on RUN QUERY:



Click on SHOW button:

