Vladimir Malchevskiy

Sprint: Микросервисная архитектура

Оглавление

1 Репозитории

GitHub link	https://github.com/ecrvmal/microservices/tree/Master

2 Used Docker commands

Following Docker commands were used:

2.1 docker commands	
2.2 docker commands	
docker images	list of docker images
docker build --no-cache -t flask_app .	build image
$ docker run --name microblog -d -p 8000:5000 --rm microblog:latest	--name имя для нового контейнера. -d указывает Docker запускать контейнер в фоновом режиме. -p сопоставляет порты контейнера с портами хоста --rm удаляет контейнер после его завершения microblog:latest img_name
docker rmi	del image
docker ps	list of working containers
docker stop container_id	Stop container (not delete)
docker stop $(docker ps -a -q)	Stop all containers
docker ps -a	see stopped container
docker rm container_id	delete container
docker rm $(docker ps -a -q)	delete all containers
DOCKER-COMPOSE commands	
docker-compose build	make a container package
docker-compose up -d	run a container package -d указывает Docker запускать контейнер в фоновом режиме. иначе открыто терминальное окно
docker exec -it container_id python manage.py createsuperuser docker exec -it 31df256d3e02 python manage.py createsuperuser docker exec -it <mycontainer> bash	run linux command in container

3 Logfiles:

3.1 Logfile Docker-compose build

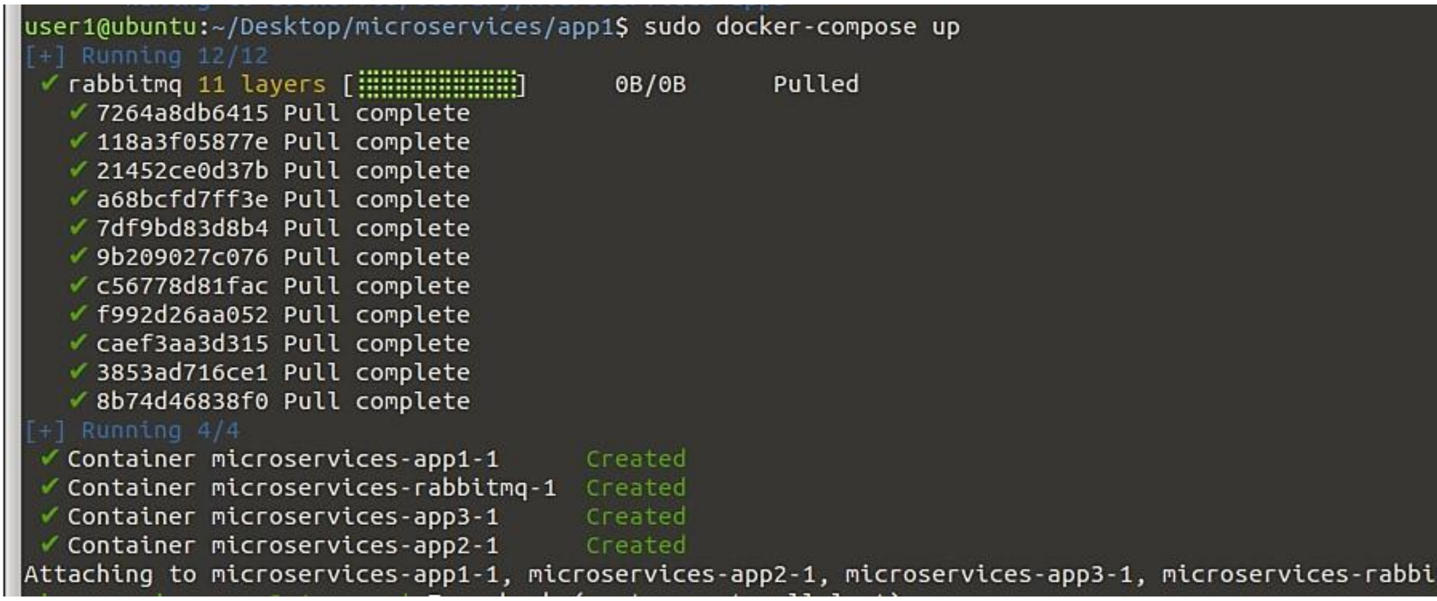
docker-compose_build.log

3.2 Logfile docker-compose up

docker-compose_up.log

4 Screenshots:

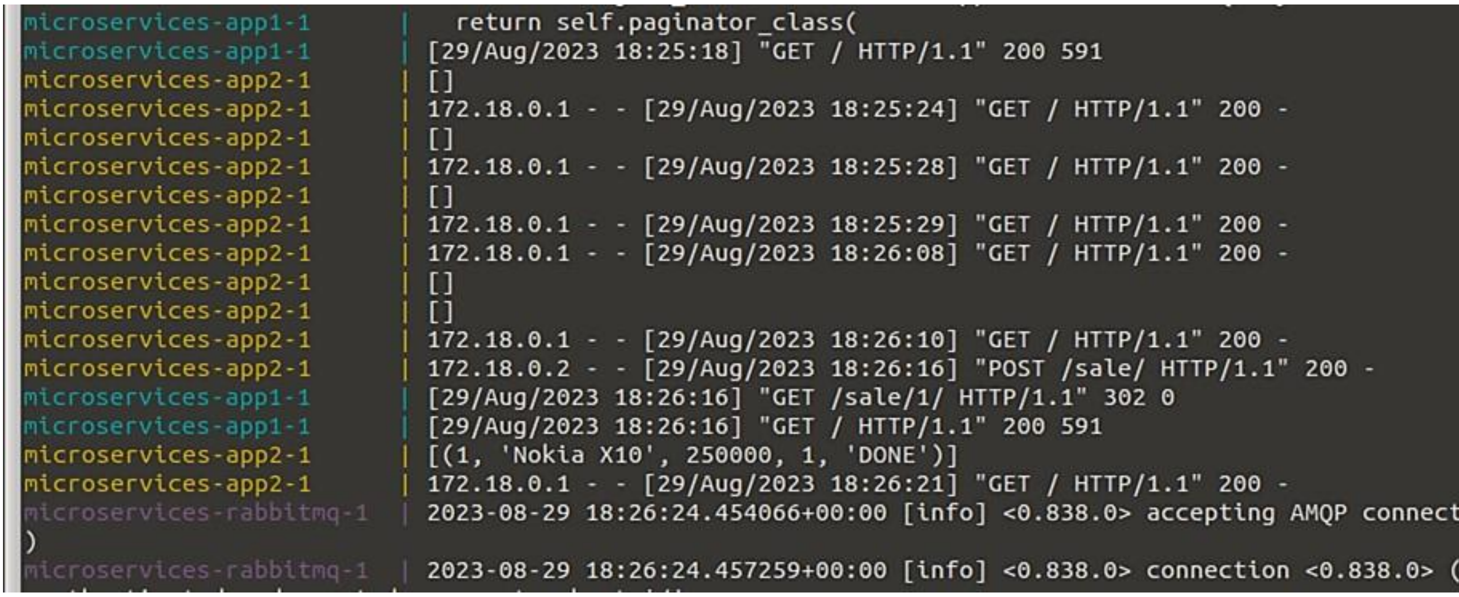
4.1 Microservices created:



```
user1@ubuntu:~/Desktop/microservices/app1$ sudo docker-compose up
[+] Running 12/12
 ✓ rabbitmq 11 layers [⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿⣿]      0B/0B      Pulled
   ✓ 7264a8db6415 Pull complete
   ✓ 118a3f05877e Pull complete
   ✓ 21452ce0d37b Pull complete
   ✓ a68bcfd7ff3e Pull complete
   ✓ 7df9bd83d8b4 Pull complete
   ✓ 9b209027c076 Pull complete
   ✓ c56778d81fac Pull complete
   ✓ f992d26aa052 Pull complete
   ✓ caef3aa3d315 Pull complete
   ✓ 3853ad716ce1 Pull complete
   ✓ 8b74d46838f0 Pull complete
[+] Running 4/4
 ✓ Container microservices-app1-1      Created
 ✓ Container microservices-rabbitmq-1  Created
 ✓ Container microservices-app3-1      Created
 ✓ Container microservices-app2-1      Created
Attaching to microservices-app1-1, microservices-app2-1, microservices-app3-1, microservices-rabbi
```

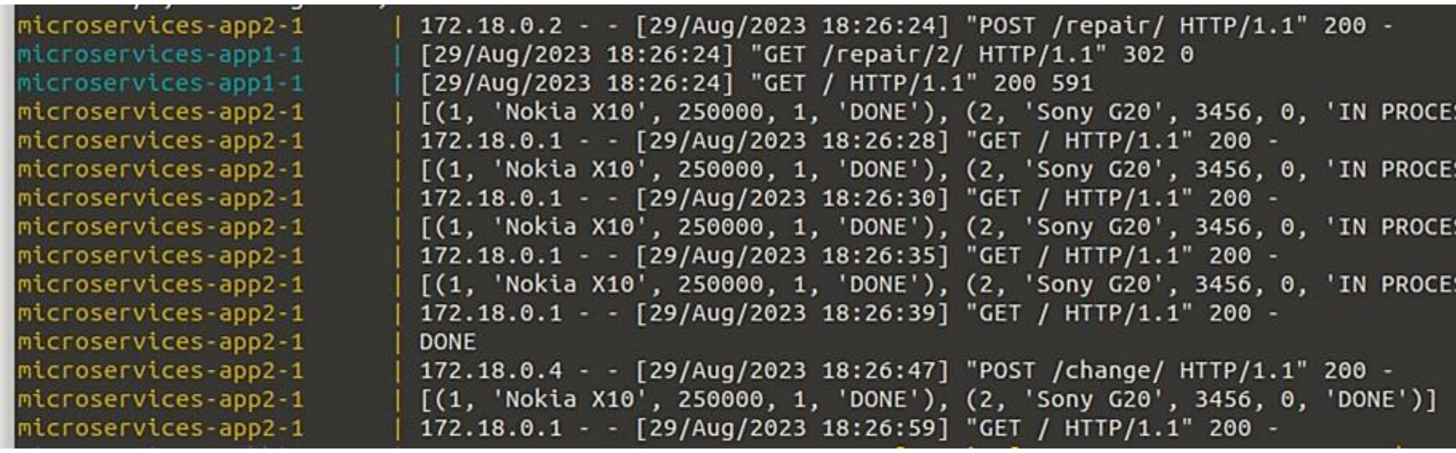
Figure 1

4.2 Microservices communicate to each other



```
microservices-app1-1       |   return self.paginator_class(
microservices-app1-1       | [29/Aug/2023 18:25:18] "GET / HTTP/1.1" 200 591
microservices-app2-1       | []
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:25:24] "GET / HTTP/1.1" 200 -
microservices-app2-1       | []
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:25:28] "GET / HTTP/1.1" 200 -
microservices-app2-1       | []
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:25:29] "GET / HTTP/1.1" 200 -
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:08] "GET / HTTP/1.1" 200 -
microservices-app2-1       | []
microservices-app2-1       | []
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:10] "GET / HTTP/1.1" 200 -
microservices-app2-1       | 172.18.0.2 - - [29/Aug/2023 18:26:16] "POST /sale/ HTTP/1.1" 200 -
microservices-app1-1       | [29/Aug/2023 18:26:16] "GET /sale/1/ HTTP/1.1" 302 0
microservices-app1-1       | [29/Aug/2023 18:26:16] "GET / HTTP/1.1" 200 591
microservices-app2-1       | [(1, 'Nokia X10', 250000, 1, 'DONE')]
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:21] "GET / HTTP/1.1" 200 -
microservices-rabbitmq-1   | 2023-08-29 18:26:24.454066+00:00 [info] <0.838.0> accepting AMQP connect
)
microservices-rabbitmq-1   | 2023-08-29 18:26:24.457259+00:00 [info] <0.838.0> connection <0.838.0> (
```

Figure 2

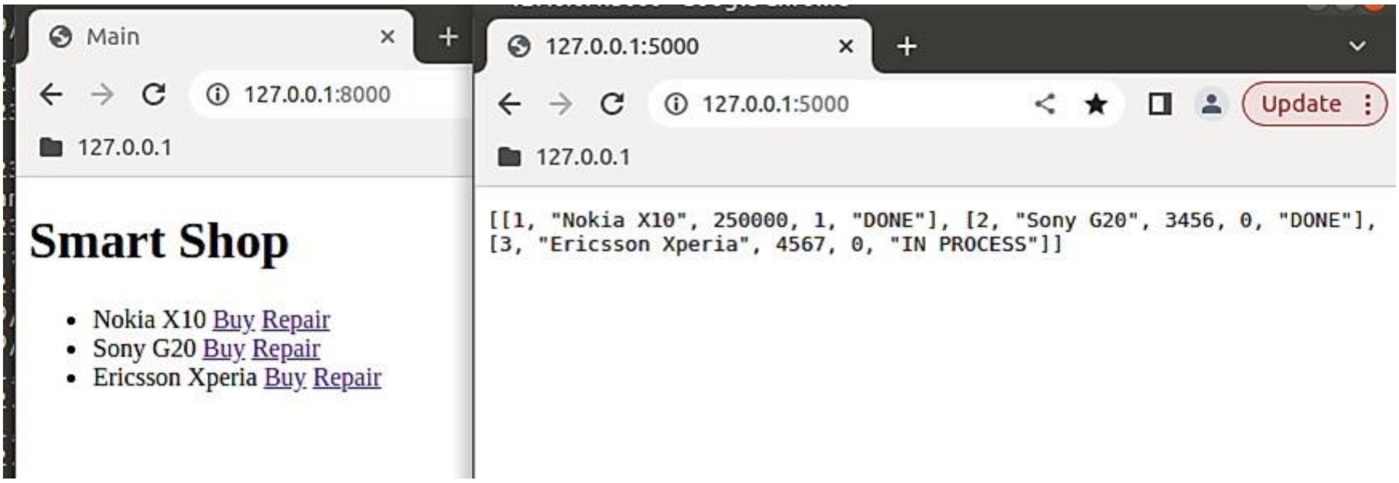


```
microservices-app2-1       | 172.18.0.2 - - [29/Aug/2023 18:26:24] "POST /repair/ HTTP/1.1" 200 -
microservices-app1-1       | [29/Aug/2023 18:26:24] "GET /repair/2/ HTTP/1.1" 302 0
microservices-app1-1       | [29/Aug/2023 18:26:24] "GET / HTTP/1.1" 200 591
microservices-app2-1       | [(1, 'Nokia X10', 250000, 1, 'DONE'), (2, 'Sony G20', 3456, 0, 'IN PROCES
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:28] "GET / HTTP/1.1" 200 -
microservices-app2-1       | [(1, 'Nokia X10', 250000, 1, 'DONE'), (2, 'Sony G20', 3456, 0, 'IN PROCE
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:30] "GET / HTTP/1.1" 200 -
microservices-app2-1       | [(1, 'Nokia X10', 250000, 1, 'DONE'), (2, 'Sony G20', 3456, 0, 'IN PROCE
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:35] "GET / HTTP/1.1" 200 -
microservices-app2-1       | [(1, 'Nokia X10', 250000, 1, 'DONE'), (2, 'Sony G20', 3456, 0, 'IN PROCE
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:39] "GET / HTTP/1.1" 200 -
microservices-app2-1       | DONE
microservices-app2-1       | 172.18.0.4 - - [29/Aug/2023 18:26:47] "POST /change/ HTTP/1.1" 200 -
microservices-app2-1       | [(1, 'Nokia X10', 250000, 1, 'DONE'), (2, 'Sony G20', 3456, 0, 'DONE')]
microservices-app2-1       | 172.18.0.1 - - [29/Aug/2023 18:26:59] "GET / HTTP/1.1" 200 -
```

Figure 3

4.3 Browser screenshots

4.3.1 Communication app1 to app2 via HTTP



Smart Shop

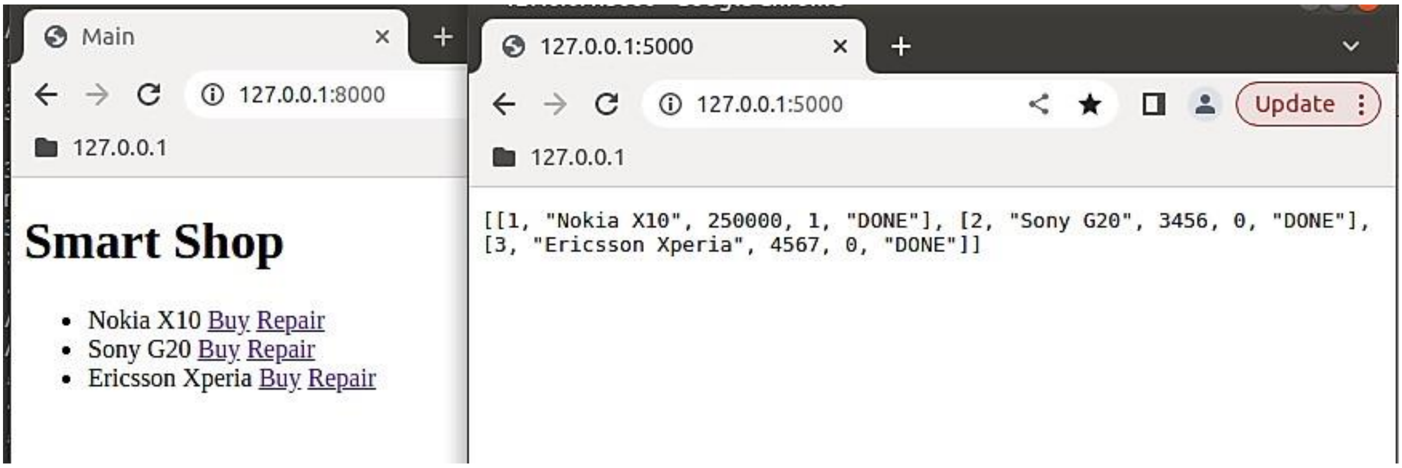
- Nokia X10 Buy Repair
- Sony G20 Buy Repair
- Ericsson Xperia Buy Repair

```
[[1, "Nokia X10", 250000, 1, "DONE"], [2, "Sony G20", 3456, 0, "DONE"],
[3, "Ericsson Xperia", 4567, 0, "IN PROCESS"]]
```

Figure 4

4.3.2 Communication app3 to app2

App3 returns status DONE



Smart Shop

- Nokia X10 Buy Repair
- Sony G20 Buy Repair
- Ericsson Xperia Buy Repair

```
[[1, "Nokia X10", 250000, 1, "DONE"], [2, "Sony G20", 3456, 0, "DONE"],
[3, "Ericsson Xperia", 4567, 0, "DONE"]]
```

Figure 5