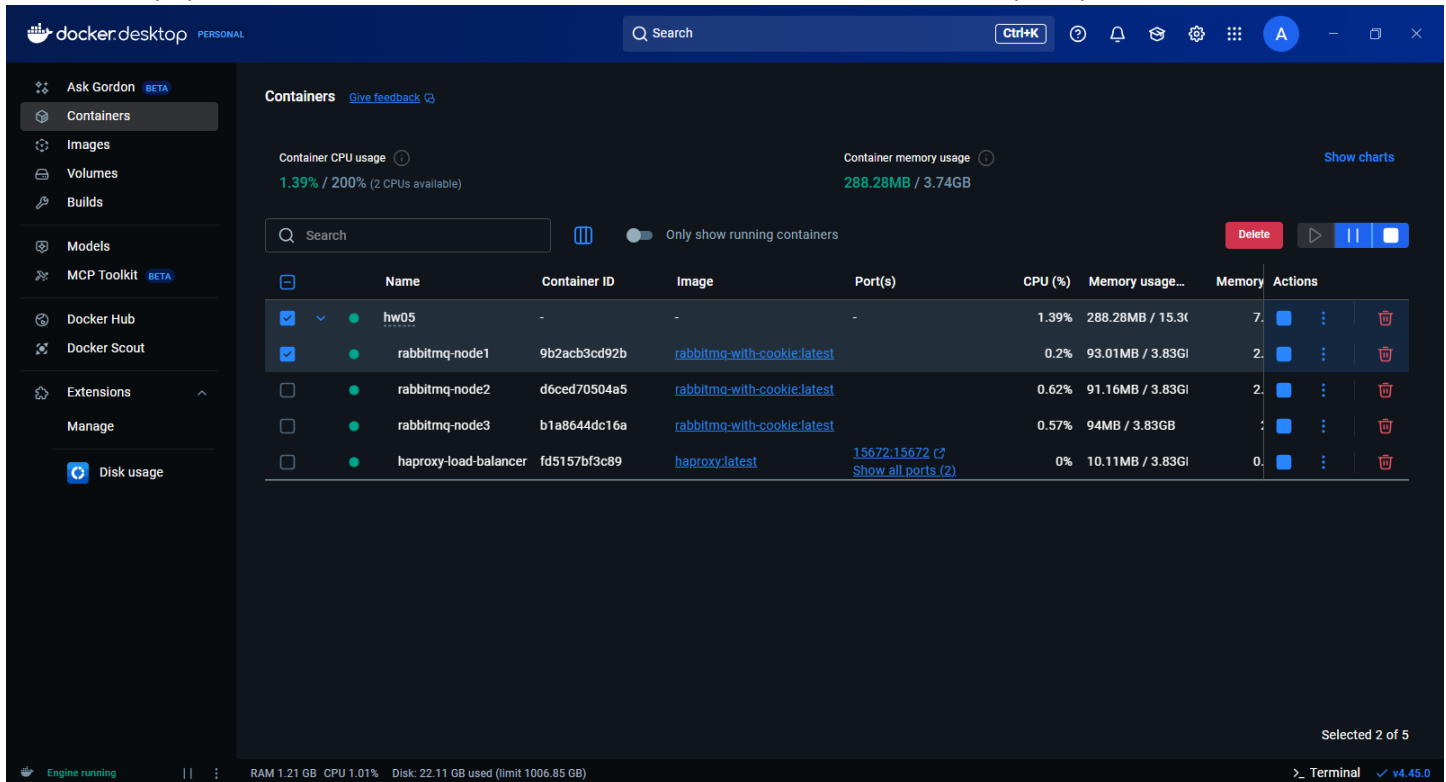


2025-07-OTUS-RABBITMQ

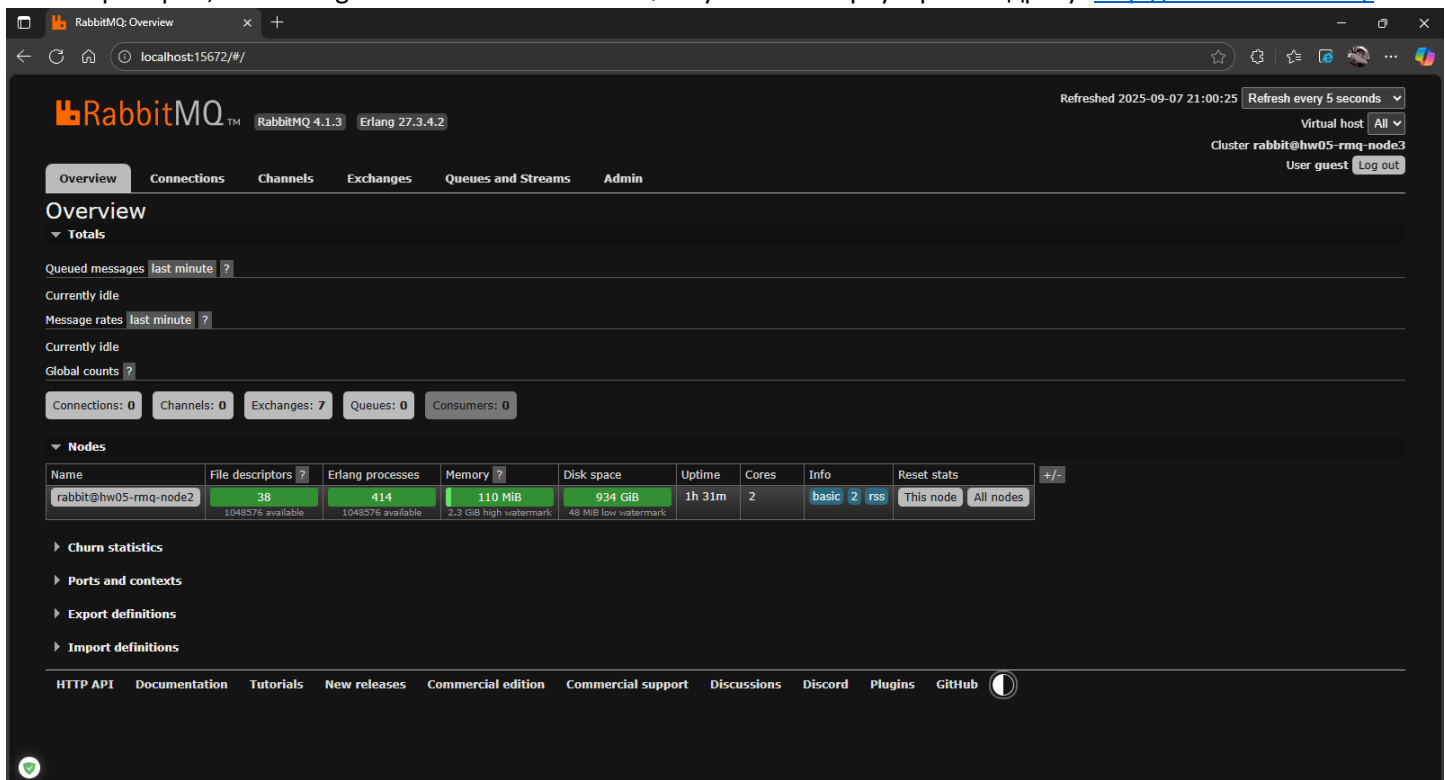
Домашнее задание №5

Александр Николаев

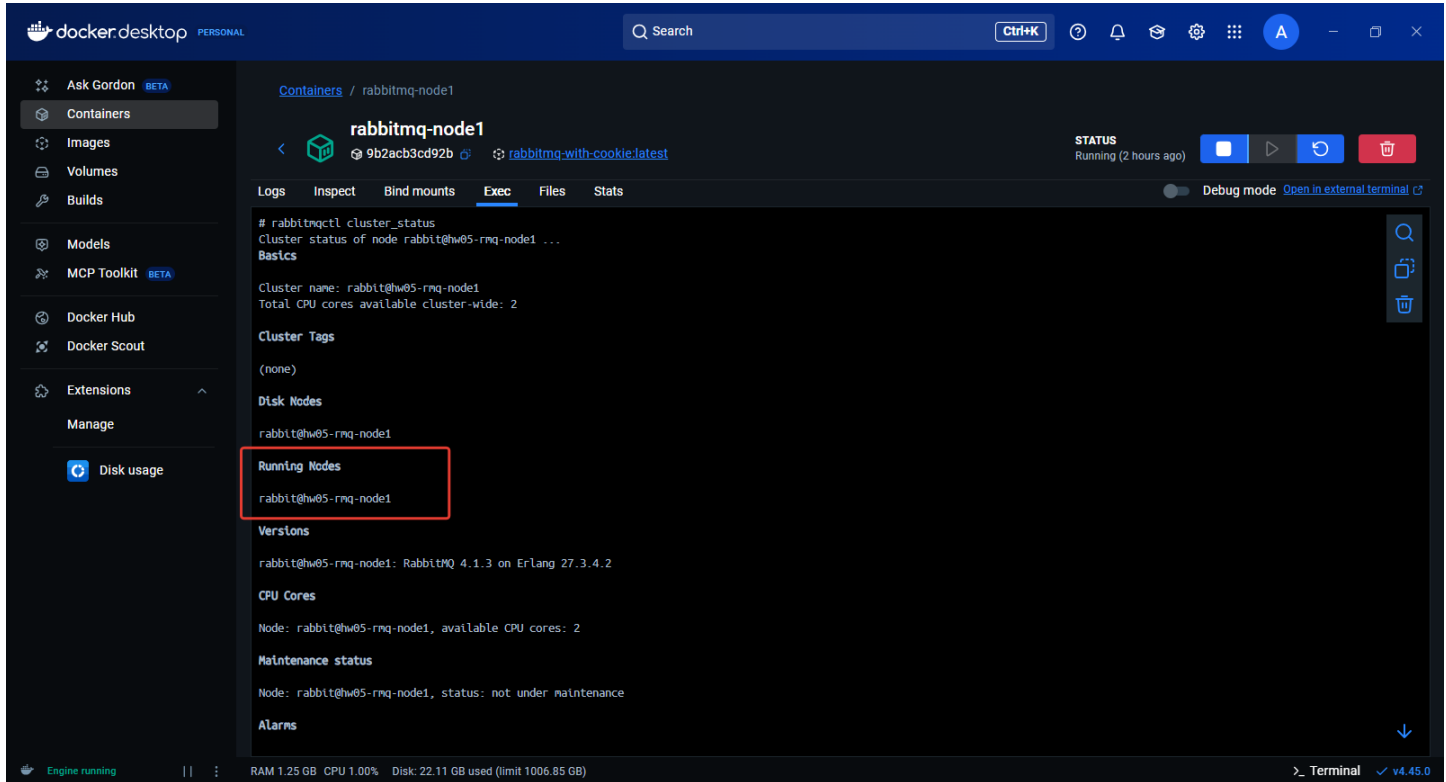
1. Создал Dockerfile для кастомизации стандартного image RabbitMQ. Заранее скопировал файл .erlang.cookie в свой image.
2. Развернул 3 независимых ноды RabbitMQ в docker с помощью docker-compose.yaml



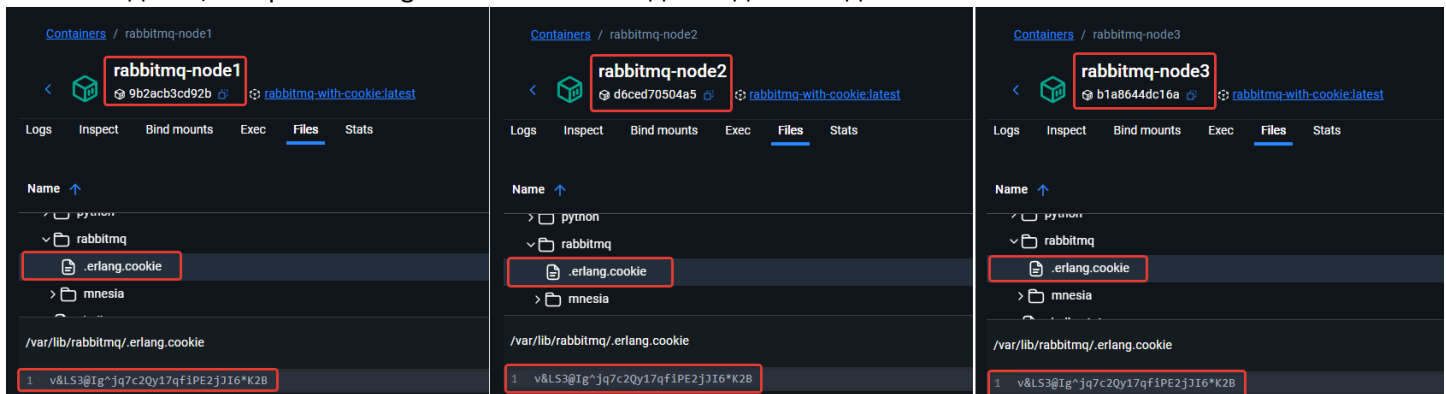
3. Проверил, что management console RabbitMQ запускается в браузере по адресу: <http://localhost:15672/>.



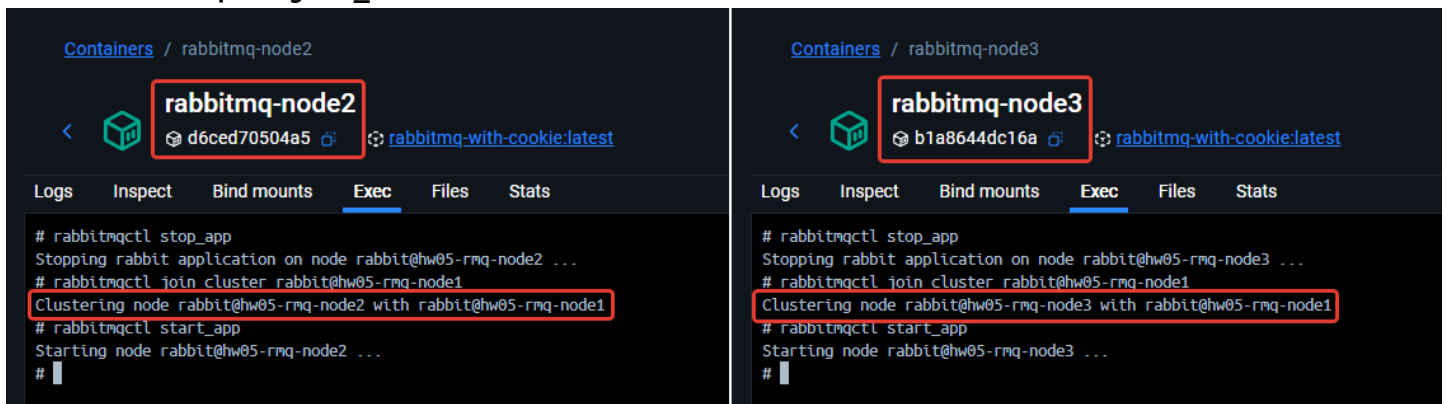
4. Убедился в том, что кластер не собран и состоит из одной ноды. Запустил команду “**rabbitmqctl cluster_status**” на ноде, которая в дальнейшем будет выступать в качестве master ноды кластера.



5. Убедился, что файл .erlang.cookie есть на каждой ноде и он одинаковый.



6. Добавляю ноды rabbitmq-node2 и rabbitmq-node3 последовательно в кластер, используя команду “**rabbitmqctl join_cluster**”.



7. Убедился в том, что кластер собран и состоит из трех нод.

RabbitMQ Overview

Refreshed 2025-09-07 22:17:22

Virtual host: All

Cluster: rabbit@hw05-rmq-node2

User: guest

Log out

Overview

Totals

Queued messages: last minute

Currently idle

Message rates: last minute

Currently idle

Global counts

Connections: 0 Channels: 0 Exchanges: 7 Queues: 0 Consumers: 0

Nodes

Name	File descriptors	Erlang processes	Memory	Disk space	Uptime	Cores	Info	Reset stats
rabbit@hw05-rmq-node1	39 1048576 available	414 1048576 available	111 MiB 2.3 GiB high watermark	934 GiB 48 MiB low watermark	2h 48m	2	basic 2 rss	This node All nodes
rabbit@hw05-rmq-node2	40 1048576 available	414 1048576 available	113 MiB 2.3 GiB high watermark	934 GiB 48 MiB low watermark	2h 48m	2	basic 2 rss	This node All nodes
rabbit@hw05-rmq-node3	39 1048576 available	412 1048576 available	113 MiB 2.3 GiB high watermark	934 GiB 48 MiB low watermark	2h 48m	2	basic 2 rss	This node All nodes

Churn statistics

Ports and contexts

Export definitions

Import definitions

HTTP API Documentation Tutorials New releases Commercial edition Commercial support Discussions Discord Plugins GitHub

8. И через командную строку, "rabbitmqctl cluster_status".

Docker Desktop

Containers / rabbitmq-node1

rabbitmq-node1

9b2acb3cd92b rabbitmq-with-cookie:latest

STATUS: Running (3 hours ago)

Logs Inspect Bind mounts Exec Files Stats

Debug mode Open in external terminal

```
# rabbitmqctl cluster_status
Cluster status of node rabbit@hw05-rmq-node1 ...
Basics
Cluster name: rabbit@hw05-rmq-node1
Total CPU cores available cluster-wide: 6
Cluster Tags
(none)
Disk Nodes
rabbit@hw05-rmq-node1
rabbit@hw05-rmq-node2
rabbit@hw05-rmq-node3
Running Nodes
rabbit@hw05-rmq-node1
rabbit@hw05-rmq-node2
rabbit@hw05-rmq-node3
Versions
rabbit@hw05-rmq-node1: RabbitMQ 4.1.3 on Erlang 27.3.4.2
rabbit@hw05-rmq-node2: RabbitMQ 4.1.3 on Erlang 27.3.4.2
rabbit@hw05-rmq-node3: RabbitMQ 4.1.3 on Erlang 27.3.4.2
CPU Cores
Node: rabbit@hw05-rmq-node1, available CPU cores: 2
Node: rabbit@hw05-rmq-node2, available CPU cores: 2
```

Engine running RAM 1.26 GB CPU 3.00% Disk: 22.11 GB used (limit 1006.85 GB)

Terminal v4.45.0

9. Создаем 3 очереди с типом "quorum" по количеству нод кластера.

```
PS D:\OTUS\RabbitMQ\2025-07-otus-rabbitmq-nikolaev> .\rabbitmqadmin --vhost "/" --node "rabbit@hw05-rmq-node1" queues declare --name "mirror-queue-node1" --type "quorum" --durable true --auto-delete false
PS D:\OTUS\RabbitMQ\2025-07-otus-rabbitmq-nikolaev> .\rabbitmqadmin --vhost "/" --node "rabbit@hw05-rmq-node2" queues declare --name "mirror-queue-node2" --type "quorum" --durable true --auto-delete false
PS D:\OTUS\RabbitMQ\2025-07-otus-rabbitmq-nikolaev> .\rabbitmqadmin --vhost "/" --node "rabbit@hw05-rmq-node3" queues declare --name "mirror-queue-node3" --type "quorum" --durable true --auto-delete false
PS D:\OTUS\RabbitMQ\2025-07-otus-rabbitmq-nikolaev> |
```

10. Проверяем что очереди создались

RabbitMQ 4.1.3 Erlang 27.3.4.2

Overview Connections Channels Exchanges **Queues and Streams** Admin

Queues

▼ All queues (3)

Pagination

Page 1 of 1 - Filter: ☐ Regex ?

Displaying 3 items, page size up to: 100

Overview							Messages			Message rates			
Virtual host	Name	Node	Type	Features	State		Ready	Unacked	Total	incoming	deliver / get	ack	+/-
/	mirror-queue-node1	rabbit@hw05-rmq-node1	quorum	D Args	running		0	0	0				
/	mirror-queue-node2	rabbit@hw05-rmq-node3	quorum	D Args	running		0	0	0				
/	mirror-queue-node3	rabbit@hw05-rmq-node1	quorum	D Args	running		0	0	0				

► Add a new queue

HTTP API Documentation Tutorials New releases Commercial edition Commercial support Discussions Discord Plugins GitHub

11. Проверяем статус кворума

docker desktop PERSONAL

Containers / rabbitmq-node1

rabbitmq-node1

9b2acb3cd92b rabbitmq-with-cookie:latest

STATUS Running (16 hours ago)

Logs Inspect Bind mounts **Exec** Files Stats

Debug mode Open in external terminal

```
# rabbitmq-queues quorum_status --vhost "/" "mirror-queue-node1"
Status of quorum queue mirror-queue-node1 on node rabbit@hw05-rmq-node1 ...
```

Node Name	Raft State	Membership	Last Log Index	Last Written	Last Applied	Commit Index	Snapshot Index	Term	Machine Version
rabbit@hw05-rmq-node1	leader	voter	2	2	2	2	-1	1	7
rabbit@hw05-rmq-node2	follower	voter	2	2	2	2	-1	1	7
rabbit@hw05-rmq-node3	follower	voter	2	2	2	2	-1	1	7

#

Engine running RAM 1.27 GB CPU 1.99% Disk: 22.11 GB used (limit 1006.85 GB)

Terminal v4.45.0

12. Проверяем, являются ли ноды критичными для кворума любых очередей

Containers / rabbitmq-node1

rabbitmq-node1

9b2acb3cd92b rabbitmq-with-cookie:latest

Logs Inspect Bind mounts **Exec** Files Stats

```
# rabbitmq-queues check_if_node_is_quorum_critical
Checking if node rabbit@hw05-rmq-node1 is critical for quorum of any queues/streams ...
Node rabbit@hw05-rmq-node1 reported no queues/streams with minimum quorum
```

Containers / rabbitmq-node2

rabbitmq-node2

d6ced70504a5 rabbitmq-with-cookie:latest

Logs Inspect Bind mounts **Exec** Files Stats

```
# rabbitmq-queues check_if_node_is_quorum_critical
Checking if node rabbit@hw05-rmq-node2 is critical for quorum of any queues/streams ...
Node rabbit@hw05-rmq-node2 reported no queues/streams with minimum quorum
```

Containers / rabbitmq-node3

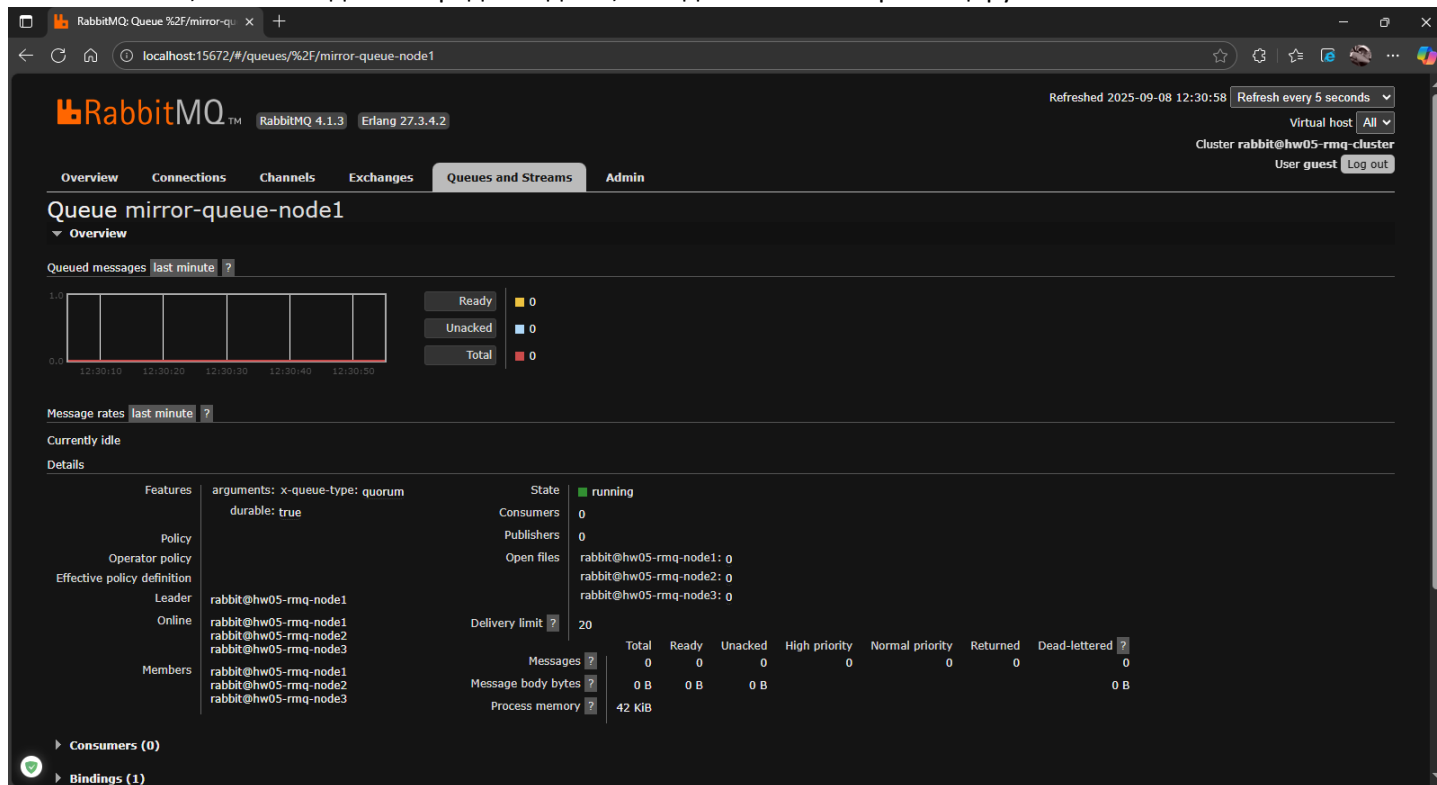
rabbitmq-node3

b1a8644dc16a rabbitmq-with-cookie:latest

Logs Inspect Bind mounts **Exec** Files Stats

```
# rabbitmq-queues check_if_node_is_quorum_critical
Checking if node rabbit@hw05-rmq-node3 is critical for quorum of any queues/streams ...
Node rabbit@hw05-rmq-node3 reported no queues/streams with minimum quorum
```

13. Считаю, что все ноды и очереди созданы, находятся онлайн и реплицируются



Выводы.

Т.к. я использовал последнюю версию RabbitMQ (4.1.3 на текущий момент), то создать классические зеркальные очереди не представляется возможным. Начиная с 4.0 версии RabbitMQ этот функционал полностью удален и взамен предлагается использовать кворумные очереди. В рамках выполнения домашнего задания я их и использовал.