name: mongo-sharding-repl

services:

configSrv:

image: mongo:latest

container\_name: configSrv

restart: always

ports:

- "27017:27017"

volumes:

- config-data:/data/db

command:

[

"--configsvr",

"--replSet", "config\_server",

"--bind\_ip\_all",

"--port", "27017"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

shard1:

image: mongo:latest

container\_name: shard1

restart: always

ports:

- "27018:27018"

volumes:

- shard1-data:/data/db

command:

[

"--shardsvr",

"--replSet", "shard1",

"--bind\_ip\_all",

"--port", "27018"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

shard2:

image: mongo:latest

container\_name: shard2

restart: always

ports:

- "27019:27019"

volumes:

- shard2-data:/data/db

command:

[

"--shardsvr",

"--replSet", "shard2",

"--bind\_ip\_all",

"--port", "27019"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

mongos\_router:

image: mongo:latest

container\_name: mongos\_router

restart: always

ports:

- "27020:27020"

command:

[

"mongos",

"--configdb", "config\_server/configSrv:27017",

"--bind\_ip\_all",

"--port", "27020"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

pymongo\_api2:

container\_name: pymongo\_api2

build:

context: api\_app

dockerfile: Dockerfile

image: kazhem/pymongo\_api1:1.0.0

depends\_on:

- mongos\_router

ports:

- 8080:8080

environment:

MONGODB\_URL: "mongodb://mongos\_router:27020" # Уточните порт для подключения

MONGODB\_DATABASE\_NAME: "somedb"

shard1\_1:

image: mongo:latest

container\_name: shard1\_1

restart: always

ports:

- "27021:27021"

volumes:

- shard1\_1-data:/data/db

command:

[

"--shardsvr",

"--replSet", "shard1",

"--bind\_ip\_all",

"--port", "27021"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

shard1\_2:

image: mongo:latest

container\_name: shard1\_2

restart: always

ports:

- "27022:27022"

volumes:

- shard1\_2-data:/data/db

command:

[

"--shardsvr",

"--replSet", "shard1",

"--bind\_ip\_all",

"--port", "27022"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

shard2\_1:

image: mongo:latest

container\_name: shard2\_1

restart: always

ports:

- "27023:27023"

volumes:

- shard2\_1-data:/data/db

command:

[

"--shardsvr",

"--replSet", "shard2",

"--bind\_ip\_all",

"--port", "27023"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

shard2\_2:

image: mongo:latest

container\_name: shard2\_2

restart: always

ports:

- "27024:27024"

volumes:

- shard2\_2-data:/data/db

command:

[

"--shardsvr",

"--replSet", "shard2",

"--bind\_ip\_all",

"--port", "27024"

]

healthcheck:

test: [ "CMD", "mongo", "--eval", "db.adminCommand('ping')" ]

interval: 5s

start\_period: 10s

volumes:

config-data:

shard1-data:

shard2-data:

shard1\_1-data:

shard1\_2-data:

shard2\_1-data:

shard2\_2-data: