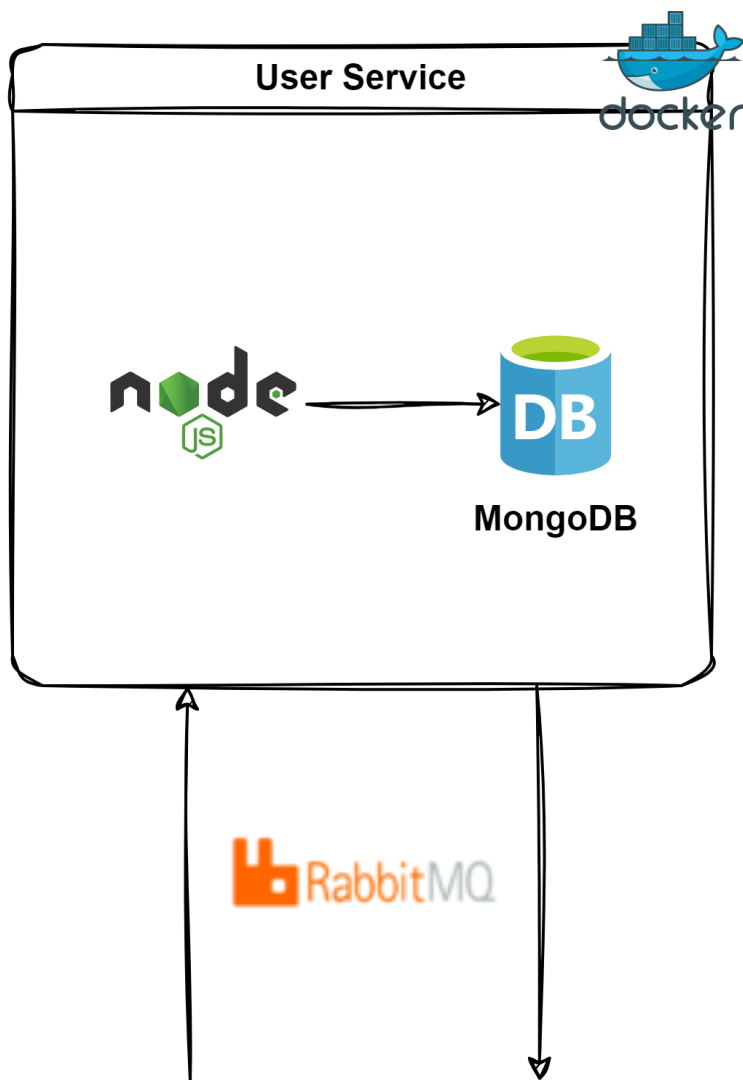


Лабораторна робота №1. Створення мікросервісу користувача.

Користувацький мікросервіс

Призначення мікросервісу:

- авторизація користувача за логіном та паролем, та повернення JWT токена у відповідь;
- надсилання повідомлень призначених для Бізнес мікросервісу до RabbitMQ;
- зчитування повідомлень з RabbitMQ від Бізнес мікросервісу.



Сценарії ініціалізації сервісу

Шаблон файлу змінних оточень

```
APPLICATION_URL=https://web.com
DB_PORT=3306
DB_USER=dbuser
```

```
DB_PASSWORD=dbpwd
DB_ROOT_PASSWORD=dbpwd
DB_NAME=dbname
DB_HOST=dbhost
JWT_SECRET_KEY=randomstring # openssl rand -base64 172 | tr -d '\n'
API_PORT=8800
NODE_LOCAL_PORT=8080
PRODUCTION=false
```

Docker-compose сервис

MongoDB

```
app-database:
  image: mongo
  restart: unless-stopped
  env_file: ../.env
  environment:
    - MONGO_INITDB_ROOT_USERNAME=$MONGODB_USER
    - MONGO_INITDB_ROOT_PASSWORD=$MONGODB_PASSWORD
  ports:
    - $MONGODB_LOCAL_PORT:$MONGODB_DOCKER_PORT
  volumes:
    - db:/data/db
```

Node.js

Dockerfile

```
ARG NODE_DOCKER_PORT
FROM node:18-alpine
RUN mkdir -p /home/node/app/node_modules && chown -R node:node /home/node/app
WORKDIR /home/node/app
COPY package*.json ./
RUN npm install
USER node
COPY --chown=node:node . .
EXPOSE $NODE_DOCKER_PORT
CMD [ "node", "server.js" ]
```

Service

```
node-app:
  build:
    context: .
  dockerfile: Dockerfile
  image: nodejs
  container_name: nodejs
  depends_on:
    - app-database
  restart: on-failure
  env_file: ../.env
  ports:
    - "$NODE_LOCAL_PORT:$API_PORT"
  environment:
    - DB_HOST=$DB_HOST
    - DB_NAME=$DB_NAME
```

```
- DB_USER=$DB_USER
- DB_PASSWORD=$DB_PASSWORD
- DB_PORT=$DB_PORT
volumes:
- ./home/node/app
- node_modules:/home/node/app/node_modules
```

Volumes

```
volumes:
db:
node_modules:
```