

ООО «Кверион»



**КВЕРИОН**

автоматизация • безопасность • связь

**«Система контроля и управления проездом автомобилей Q-YMS»**

Сокращенное наименование: Система Q-YMS

Версия 1.0

**Информация по установке программного обеспечения**

Краснодар, 2023 г.

## Содержание

### Содержание

1.	Наименование.....	3
2.	Общая информация о системе .....	3
3.	Установка среды Docker .....	5
4.	Установка и запуск системы Q-YMS .....	6

## 1. Наименование

**Полное наименование:** Система контроля и управления проездом автомобилей Q-YMS.

**Техническое наименование:** Система Q-YMS.

## 2. Общая информация о системе

Программное обеспечение Q-YMS представляет собой систему на основе веб сервисов, взаимодействующих через сетевые протоколы такие HTTP, MQTT, ModBus и другие. Основное назначение системы – это контроль передвижения транспортных средств на производственных площадках через управление исполнительными механизмами (шлагбаумы, болларды и т.п.), а также мониторинг маршрутов следования транспортных средств в зонах контролируемых системой Q-YMS. Взаимодействие сервисов и взаимосвязь между API конфигурируется через сервер потокового программирования Node-RED.

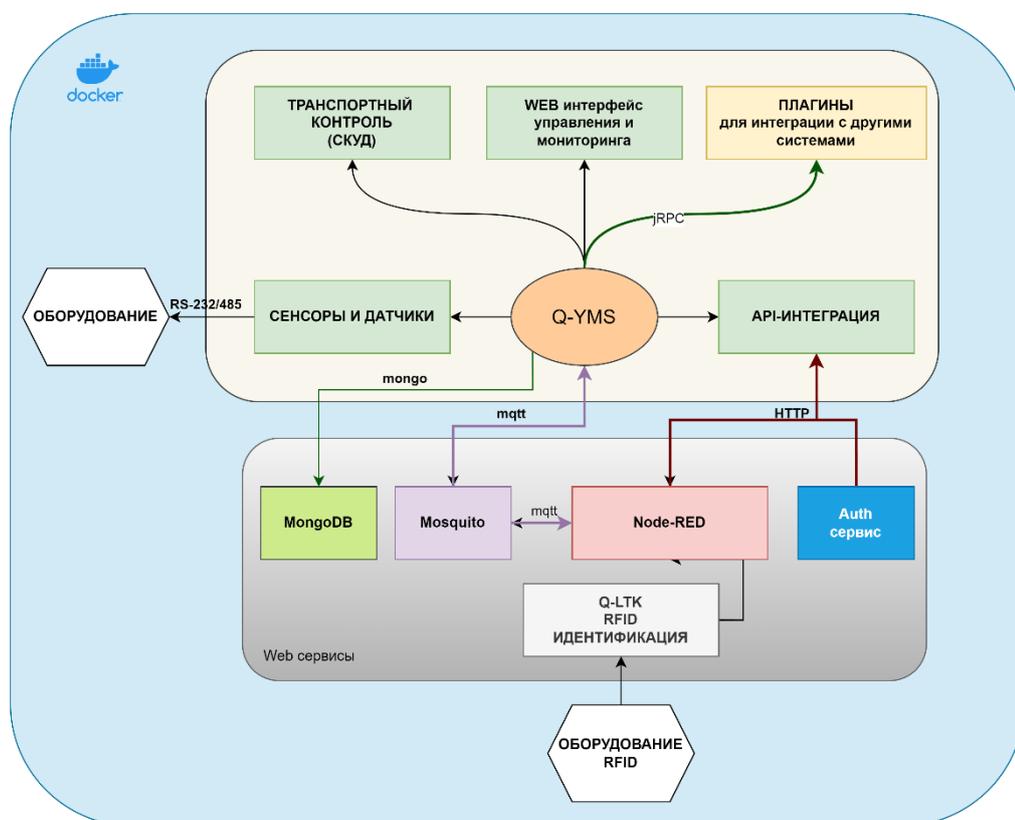


Рис.1. Архитектура системы Q-YMS

1. **Сервис Q-YMS** основной сервис управления системой, который обеспечивает взаимодействие с остальными компонентами системы.

Разворачивается в среде docker через утилиту оркестрации docker-compose. Основной файл конфигурации '**config.json**' монтируется к каталогу контейнера '**/app/conf**', файлы HTTPS сертификатов монтируются к каталогу контейнера '**/app/cert**'. Внутри контейнера сервис по умолчанию открывает порты **80** и **443**.

2. **Сервис базы данных MongoDB** – сервис основной базы данных системы Q-YMS. Разворачивается в среде docker через утилиту оркестрации docker-compose. Для инициализации учетных данных доступа и коллекции базы данных используется файл **init-mongo.js**. Директория для данных монтируется к внутреннему каталогу контейнера '**/data/db**'. Внутри контейнера сервис, по умолчанию, открывает порты: '**27017**'.

3. **Сервис Mosquitto** – основной брокер шины обмена данными между сервисами по протоколу MQTT с модулем авторизации через Q-YMS. Разворачивается в среде docker через утилиту оркестрации docker-compose. Директория для конфигурации, данных монтируется к внутреннему каталогу контейнера «**/etc/mosquitto/**». Внутри контейнера сервис, по умолчанию, открывает порты: '**1883**', '**9001**'.

4. **Сервис Node-RED** – используется для визуального построения алгоритмов в соответствии с бизнес-процессами заказчика. Разворачивается в среде docker через утилиту оркестрации docker-compose. Директория для данных монтируется к внутреннему каталогу контейнера '**/data**'. Внутри контейнера сервис, по умолчанию, открывает порты: '**1880**', '**162**'

5. **Сервис Q-LTK** – используется для взаимодействия с RFID считывателями по протоколу LLRP. Разворачивается в среде docker через утилиту оркестрации docker-compose. Директория для конфигурации монтируется к внутреннему каталогу контейнера '**/data**'. Сервис взаимодействует с другими подсистемами через Webhook вызовы.

6. **Сервис Auth** – позволяет выполнять авторизацию внешних сервисов через подсистему Q-YMS. Разворачивается в среде docker через утилиту оркестрации docker-compose. Директория для конфигурации монтируется к внутреннему каталогу контейнера '**/conf**'. Внутри контейнера сервис по умолчанию открывает порты **80** и **443**.

#### **Системные требования:**

- Поддерживаемые операционные системы:
  - Linux Debian x64 (среда Docker) версия 9,10,11;
- Процессор: архитектура x64, тактовой частотой не менее 2,2 GHz;
- Оперативная память: не менее 8 Gb;
- Сетевой адаптер: 1 Gbit/s;
- Система хранения: не менее 200 Gb (в зависимости от объема данных). Рекомендуется 1TB;
- СУБД: MongoDB (версия 5.0);

### 3. Установка среды Docker

1. Необходимо произвести настройка **apt** репозитория для установки пакетов среды **Docker Engine CE**.

```
# Добавление ключа для проверки подписи GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl gnupg
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg

# Добавление источника данных для репозитория Docker:
echo \
  "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg]
  https://download.docker.com/linux/debian \
  "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

2. Установка пакетов для **Docker Engine CE**

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

3. Проверка корректности установки и работы среды **Docker Engine CE**.

```
sudo docker run hello-world
```

Контейнер должен запуститься и выдать сообщение о том, что система работает корректно.

4. Выполнить авторизацию в приватный реестр docker **'registry.systems-it.ru'**.

**Данные для авторизации предоставляются по запросу клиента.**

```
sudo docker login https://registry.systems-it.ru:5000
```

## 4. Установка и запуск системы Q-YMS

1. Для установки системы необходимо создать каталог `'/opt/q-yms'`

```
sudo mkdir /opt/q-yms
```

2. Загрузить в него архив с дистрибутивом Q-YMS с подготовленной средой контейнеризации Docker. Дистрибутив доступен по ссылке

<https://querion.ru/qyms/dist.zip>. Архив представляет собой

⚠ ПАРОЛЬ ОТ АРХИВА С ДИСТРИБУТИВОМ ПРЕДОСТАВЛЯЕТСЯ ТОЛЬКО ПО ЗАПРОСУ КЛИЕНТА

предустановленный стек для запуска с использованием **docker-compose**

3. Распакуйте содержимое архива **'dist.zip'**. в созданный в п.1 каталог.

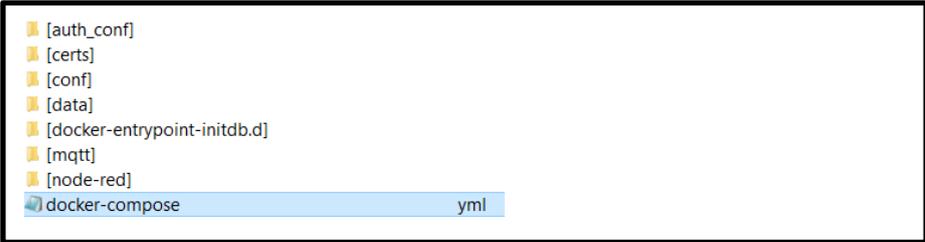


Рис.2. Пример содержимого каталога `'/opt/q-yms'`

4. В каталоге, где расположен файл **'docker-compose.yml'**, необходимо выполнить команду:

```
sudo docker compose up -d
```

⚠ Для версии Docker Engine ниже 20.x утилита docker-compose устанавливается отдельно

5. После успешного запуска к системе можно получить доступ:

**Q-YMS** - `http://{docker-host}:55580`

Логин: admin

Пароль: admin

**Node-RED** - `http://{docker-host}:51880/node-red`

Логин: admin

Пароль: password

где: **{docker-host}** - это реальное имя хоста или IP-адрес на котором развернута среда Docker и запущен стек Q-YMS.