**Инструкция**

**по установке экземпляра программного обеспечения «Программный комплекс для дистанционного банковского обслуживания»**

Доступ к демо-версии программного обеспечения (песочнице)

Адрес: <https://cube.dev.self.team>

**Данные для первичной аутентификации в браузере:**

логин: devuser

пароль: Xq91kL87tG3$6h

**Данные для входа в систему:**

Логин для входа: +7(986) 770-88-02

Пароль при входе: 123456

Одноразовый пароль: 1234

Контактные данные службы поддержки: [support@self.team](mailto:support@self.team)

+7 495 136 62 71,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Оркестровщик АРМ запросов

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t admin:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 admin:latest --name admin --env-file .env

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| USERBOX\_HOST | Адрес компонента UserBox | string | + | |

| ADMIN\_BUTLER\_HOST | Адрес компонента Butler (Администраторы) | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| BANK\_LIMIT\_HOST | Адрес компонента BankLimit | string | + | |

| POSTIE\_HOST | Адрес компонента Postie | string | + | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t bank-limit:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 bank-limit:latest --name bank-limit --env-file .env

```

Инициализировать структуру БД.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti bank-limit bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | bank-limit |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | pgsql |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | 5432 |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Микросервис банковских продуктов пользователя

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t bank-product:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 bank-product:latest --name bank-product --env-file .env

```

Инициализировать структуру БД и наполнить данными.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti bank-product bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate && php artisan db:seed --class BasicSeeder --force"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

| USERBOX\_HOST | Адрес компонента UserBox | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Selfteam Bus Manager

Микросервис может выполянет две группы задач:

- Отправляет сообщения в очереди и возвращает полученный ответ

- [ ] Синхронно - клиент передает сообщение и сразу получает ответ

- [ ] Асинхронно - клиент передает сообщение и отдельным запросом получает ответ

- Слушает внешние очереди, парсит запрос и складывает в нашу БД для последующей обработки

- [ ] Прочитал и ничего не делает

- [ ] Прочитал сообщение и знает как обаработать это сообщение

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t busmanager:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 busmanager:latest --name busmanager --env-file .env

```

Инициализировать структуру БД и наполнить данными.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti busmanager bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | busmanager |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | pgsql |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | 5432 |

| DB\_DATABASE | Название базы данных | string | + | busmanager |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| SALT | Соль | string | + | |

| LOG\_MESSAGES\_IN\_DB | Сохранять исходящие и входящие сообщения в БД | bool | - | false |

| | | | | |

| \_# Remote Components\_ |-|-|-|-|

| USERBOX\_HOST | Адрес компонента UserBox | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| BANK\_PRODUCT\_HOST | Адрес компонента BankProduct | string | + | |

| BANK\_LIMIT\_HOST | Адрес компонента BankLimit | string | + | |

| REGMANAGER\_HOST | Адрес компонента RegManager | string | + | |

| CURRENCY\_HOST | Адрес компонента Currency | string | + | |

| USER\_SETTINGS\_HOST | Адрес компонента UserSettings | string | + | |

| PAYMENTGATE\_HOST | Адрес компонента PaymentGate | string | + | |

| APIGATE\_URL | Адрес API-GATE | string | + | |

| | | | | |

| \_# MQ Server\_ |-|-|-|-|

| MQ\_QUEUE\_MANAGER | Менеджер очередей (?) | string | + | WMB01TQM |

| MQ\_CHANNEL | Канал (?) | string | + | DBO.MQ.SVRCONN |

| MQ\_HOST | Адрес очереди | | | |

| MQ\_PORT | Порт очереди | | | |

| MQ\_MAX\_MSG\_LENGTH | Максимальная длина сообщения в очереди в байтах (?) | string | - | 4194304 |

| MQ\_DEFAULT\_WAIT\_INTERVAL | Время ожидания ответа от очереди (?) | string | - | 60000 |

| MQCCSID | ? | string | + | 1208 |

| MQCCRED | ? | string | + | /opt/mqm/mqccred.ini |

| MQCHLLIB | ? | string | + | /opt/mqm/ |

| MQCHLTAB | ? | string | + | AMQCLCHL.TAB |

| \_# XSD\_ |-|-|-|-|

| ENABLE\_XSD\_VALIDATION | Вкл/выкл валидации xsd (?) | boolean | - | false |

| | | | | |

| \_# Очереди, где инициатором выступает ДБО\_ | | | | |

| MQ\_DBO\_\_DBO\_TO\_WS\_PRIM | Очередь приоритетных запросов от ДБО к шине | string | + | |

| MQ\_DBO\_\_WS\_TO\_DBO\_PRIM | Очередь ответов на приоритетные запросы от ДБО к шине | string | + | |

| MQ\_DBO\_\_DBO\_TO\_WS\_SEC | Очередь неприоритетных запросов от ДБО к шине | string | + | |

| MQ\_DBO\_\_WS\_TO\_DBO\_SEC | Очередь ответов на неприоритетные запросы от ДБО к шине | string | | |

| MQ\_DBO\_\_DBO\_TO\_CFT | Очередь запросов из ДБО в АБС ЦФТ | string | + | |

| MQ\_DBO\_\_CFT\_TO\_DBO | Очередь ответов АБС ЦФТ на запросы из ДБО с сформированным WebSphere CorelId | string | + | |

| MQ\_DBO\_\_DBO\_TO\_SBL | Очередь запросов из ДБО в Siebel | string | + | |

| MQ\_DBO\_\_SBL\_TO\_DBO | Очередь ответов ДБО на запросы из Siebel | string | + | |

| MQ\_DBO\_\_DBO\_TO\_CUP | Очередь сообщений из ДБО в ЦУП | string | + | |

| MQ\_DBO\_\_CUP\_TO\_DBO | Очередь сообщений из ЦУП в ДБО | string | + | |

| MQ\_DBO\_\_DBO\_TO\_CCC | Очередь запросов из ДБО в TripleC | string | + | |

| MQ\_DBO\_\_CCC\_TO\_DBO | Очередь ответов TripleC на запрос из ДБО | string | + | |

| \_## Прямое взаимодействие с зибелем\_ | | | | |

| MQ\_DBO\_\_DBO\_TO\_SBL\_DIRECT | Очередь запросов из ДБО в Siebel | string | + | |

| MQ\_DBO\_\_SBL\_TO\_DBO\_DIRECT | Очередь ответов Siebel на запросы из ДБ | string | + | |

| | | | | |

| \_# Очереди, где инициатором выступает БАНК\_ | | | | |

| MQ\_WS\_\_WS\_TO\_DBO\_PRIM | Очередь приоритетных запросов от шины к ДБО | string | + | |

| MQ\_WS\_\_DBO\_TO\_WS\_PRIM | Очередь ответов на приоритетные запросы от ДБО к шине | string | + | |

| MQ\_WS\_\_WS\_TO\_DBO\_SEC | Очередь неприоритетных запросов от шины к ДБО | string | + | |

| MQ\_WS\_\_DBO\_TO\_WS\_SEC | Очередь ответов на неприоритетные запросы от ДБО к шине | string | + | |

| MQ\_TSO\_\_TSO\_TO\_DBO\_PROXY | Очередь запросов из УПО ATM/ТСО в ДБО | string | + | |

| MQ\_TSO\_\_DBO\_TO\_TSO\_PROXY | Очередь ответов ДБО на запросы из УПО ATM/ТСО | string | + | |

| \_## Прямое взаимодействие с зибелем\_ | | | | |

| MQ\_SBL\_\_SBL\_TO\_DBO\_DIRECT | Очередь запросов из Siebel в ДБО | string | + | |

| MQ\_SBL\_\_DBO\_TO\_SBL\_DIRECT | Очередь ответов ДБО на запросы из Siebel | string | + | |

| | | | | |

| \_# Очереди, где инициатором выступает БАНК и отвечать не требуется\_ | | | | |

| MQ\_WS\_\_WS\_TO\_DBO\_ASYNC | Очередь передачи курсов валют банка для ДБО | string | + | |

| | | | | |

| \_# Очереди для работы с FormIt\_ | | | | |

| MQ\_DBO\_\_DBO\_TO\_FORM\_IT | Очередь запросов из ДБО в FormIT | string | + | |

| MQ\_DBO\_\_FORM\_IT\_TO\_DBO | Очередь ответов FormIT на запросы из ДБО | string | + | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Butler

Компонент отвечает за аутентификацию и авторизацию клиента.

Набор хранимой информации о клиенте ограничивается минимальным набором данных, необходимым для выполнения функций компонента.

Полная информация о клиента доступна в Компоненте Профиль клиента.

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t butler:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 butler:latest --name butler --env-file .env

```

Инициализировать структуру БД и наполнить данными.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti butler bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate && php artisan db:seed --class BasicSeeder --force"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

| # Настройка безопасности | | | | |

| AUTH\_BY\_LOGIN\_OTP\_DISABLED | Отключить OTP на вход по логину и паролю | bool | - | false |

| PASSWORD\_MAX\_ATTEMPTS | Максимальное число попыток ввода пароля перед блокировкой канала | int | - | 5 |

| MAX\_ACTIVE\_SESSIONS\_PER\_CHANNEL | Максимальное кол-во сессий на канал | int | - | 1 |

| ACCESS\_TOKEN\_PUBLIC\_KEY | Публичный ключ для проверки JWT | string | - | |

| ACCESS\_TOKEN\_PRIVATE\_KEY | Приватный ключ для подписывания JWT | string | - | |

| ACCESS\_TOKEN\_TTL | Время жизни сессии, в минутах | int | - | 60 |

| MAX\_ATTEMPTS\_CHANGE\_PASSWORD | Максимальное число попыток сменить пароль | int | - | 3 |

| BLOCK\_TIMEOUT\_CHANGE\_PASSWORD | Время блокировки смены пароля (в минутах) | int | - | 60 |

| | | | | |

| | | | | |

| # OTP | | | | |

| OTP\_ENABLED | Вкл/выкл подтверждения одноразовым паролем | boolean | - | true |

| OTP\_DEBUG | Вкл/выкл режима отладки OTP | boolean | - | false |

| OTP\_TTL | Время жизни токена OTP в секундах | int | - | 60 |

| OTP\_MAX\_ATTEMPTS | Максимальное число попыток ввода кода OTP | int | - | 4 |

| SMS\_CODE\_LENGTH | Длина OTP кода для SMS | int | - | 4 |

| POSTIE\_ENABLED | Вкл/выкл режим отладки для отправляемых сообщений | boolean | - | false |

| # Межкомпонентное взаимодействие | | | | |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| USE\_NEW\_SESSION\_TOGGLE | Включает использование нового механизма работы с сессиями | boolean | - | false |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# CreditMarket

Сервис для работы с кредитными предложениями.

## Artisan

@todo

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t credit-market:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 credit-market:latest --name credit-market --env-file .env

```

Инициализировать структуру БД.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti credit-market bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | credit-market |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | pgsql |

| DB\_HOST | Адрес сервера БД | string | + | 192.168.60.93 |

| DB\_PORT | Порт сервера БД | string | + | 5432 |

| DB\_DATABASE | Название базы данных | string | + | credit-market |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| USERBOX\_HOST | Адрес компонента Userbox | string | + | |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| SALT | Соль | string | + | |

| SLF\_SIEBEL\_DOWNLOAD\_LOGIN | Логин в зибель для скачивания документов | string | + | BIPUSER |

| SLF\_SIEBEL\_DOWNLOAD\_PASSWORD | Пароль в зибель для скачивания документов | string | + | 123456 |

## Клиентские методы

\*\*[POST] /v1/credit/applications/list\*\* — список кредитных предложений

\*\*[POST] /v1/credit/order/create\*\* — создание заявки на кредит

\*\*[POST] /v1/credit/order/refuse\*\* — отмена заявки на кредит

\*\*[POST] /v1/credit/order/refuse/reason/list\*\* — список причин для отмены

\*\*[POST] /v1/credit/order/documents/get\*\* — список документов по предложению

\*\*[POST] /v1/credit/order/documents/send\*\* — отправка документов по предложению на почту

\*\*[GET] /v1/credit/order/documents/download\*\* — загрузка всех документов

\*\*[GET] v1/credit/order/document/download\*\* — загрузка одного документа

\*\*[GET] /v1/credit/order/document/show\*\* — просмотр одного документа

Подробная спецификация

[здесь](<http://192.168.1.11:8098/?urls.primaryName=credit-market#/>).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Курсы валют и конвертация

## Терминология

CurrencyRate включает в себя два курса - продажу и покупку.

Рассмотрим две записи:

```

baseCurrency: USD (Валюта счета зачисления)

quoteCurrency: RUB (Валюта счета списания)

operationType: sale

rate: 62

-- Продаем USD за RUB по курсу 1 USD = 62 RUB

```

```

baseCurrency: USD

quoteCurrency: RUB

operationType: buy

rate: 65

-- Покупаем USD за RUB по курсу 65 RUB за 1 USD

```

## Artisan

``currency:rate:update`` - заполнение БД курсами валют

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t currency:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 currency:latest --name currency --env-file .env

```

Инициализировать структуру БД и наполнить данными.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti currency bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate && php artisan db:seed --class BasicSeeder --force"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

| CURRENCY\_RATE\_DATA\_SOURCE | Источник курсов валют (если одновлять через artisan) | enum: CBR | - | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Депозитный калькулятор

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t deposit-market:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 deposit-market:latest --name deposit-market --env-file .env

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| USERBOX\_HOST | Адрес компонента UserBox | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| BANK\_PRODUCT\_HOST | Адрес компонента BankProduct | string | + | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Directory

Микросервис для работы с различными справочниками

## Artisan

```

import:bic Импорт справочник БИК с сайта ЦБ

import:otp:branch Импорт спраочника филиалов банка

import:region Заполнить таблицу с регионами РФ

import:swift Заполнить таблицу SWIFT

import:swift:country Импорт стран для справочника SWIFT

seed:branch-or-atm Создать справочник отделения и банкоматов

seed:card:type Создать изображения кард по умолчанию

seed:merchant Создать фейковых мерчантов

seed:operation-category Создать справочник дефолтных иконок категорий PFM

```

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t directory:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 directory:latest --name directory --env-file .env

```

Инициализировать структуру БД и заполнить данными.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan import:bic"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan import:otp:branch"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan import:region"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan import:swift resources/swift/BICDIR2018\_V1\_FULL\_20190222.xml"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan import:swift:country resources/swift/COUNTRY\_CODE\_20190222.xml"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan seed:branch-or-atm"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan seed:card:type"

docker exec -ti directory bash -c \

"cd /var/www/app/src && php artisan seed:operation-category"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | directory |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_HOST | Домен с веб интерфейсом | string | - | |

| STATIC\_FILES\_HOST | Домен для отдачи статики | string | - | |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | pgsql |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | 5432 |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| SALT | Соль | string | + | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Сервис клиентских уведомлений

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать \*\*dockerlogs2\*\* | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию. Должно быть равно \*\*redis\*\* | string | - | file |

| REDIS\_HOST | Хост Redis | string | + | |

| REDIS\_PORT | Порт Redis | string | + | |

| DB\_CONNECTION | Драйвер подключения к БД. Должно быть равно \*\*pgsql\*\* | string | + | |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

| DOCTRINE\_CACHE | Драйвер кеша ORM. Должно быть равно \*\*chain\*\* | string | - | |

| SALT | Соль | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

## API

### Внутреннее API. Должно быть доступно только из внутренней сети

\* \*\*/v1/octopus/event/handle\*\* — обработка события из Octopus

\* \*\*/v1/octopus/event/list/supported\*\* — список поддерживаемых событий

### Публичное API

\* \*\*/v1/notification/list\*\* — список уведомлений

\* \*\*/v1/notification/count/new\*\* — счётчик новых уведомлений

\* \*\*/v1/notification/set/seen\*\* — пометить уведомление как прочитанное

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Сервис новостей Банка

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать \*\*dockerlogs2\*\* | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию. Рекомендуется указать \*\*redis\*\* | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| REDIS\_HOST | Хост Redis | string | - | |

| REDIS\_PORT | Порт Redis БД | string | - | |

| DOCTRINE\_CACHE | Драйвер кеша Doctrine. Для использования кеша приложения следует указать значение \*\*chain\*\* | string | - | array |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

| SALT | Соль | string | + | |

| BUTLER\_HOST | Адрес компонента Butler. В случае, если развёрнут экземпляр приложения для администрирования, должен быть указан адрес админского Butler | string | + | |

## Методы для администрирования

Запросы к методам администрирования должны быть направлены в

соответствующий экземпляр приложения

\*\*/v1/article/list/filtered\*\* — список всех новостей

\*\*/v1/article/create\*\* — создание новости

\*\*/v1/article/update\*\* — обновление новости

\*\*/v1/article/content/add\*\* — добавление текста

\*\*/v1/article/visibility/toggle\*\* — переключение видимости новости

\*\*/v1/article/remove\*\* — удаление новости

\*\*/v1/content/update\*\* — обновление текста

\*\*/v1/content/remove\*\* — удаление текста

\*\*/v1/action/add\*\* — добавление действия

\*\*/v1/action/update\*\* — обновление действия

\*\*/v1/action/content/add\*\* — добавления текста действия

\*\*/v1/action/remove\*\* — удаление действия

## Клиентские методы

\*\*/v1/article/list/guest\*\* — список новостей для неавторизованного

пользователя

\*\*/v1/article/list/auth\*\* — список новостей для авторизованного

пользователя

\*\*/v1/article/hide\*\* — скрыть новость

Подробная спецификация

[здесь](http://192.168.1.11:8098/?urls.primaryName=news#/).

## Установка

Приложение должно быть развёрнуто, как минимум, в двух экземплярах: один для администрирования и один для клиенстких запросов.

Все экземпляры должны работать с одним и тем же источником данных.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# SelfTeam.Octopus

Laravel-based queue manager service

## Handling queue manager

In order to start Octopus manually one should execute the following command

```bash

php /path/to/octopus/src/artisan horizon > /dev/null 2>&1 &

```

That can also be done via Monit util.

Octopus can be shut down by the following command

```bash

php /path/to/octopus/src/artisan horizon:terminate

```

## Job presets

Some jobs can be scheduled from CLI using Octopus' built-in preset feature.

```bash

# Schedule job preset

# alias — required parameter for job preset alias

# --cron — optional parameter for custom cron. example: \*/5 \* \* \* \*

php /path/to/octopus/src/artisan job:preset:schedule <alias> [--cron=<cron>]

# Cancel scheduled job

# alias — required parameter for job preset alias

php /path/to/octopus/src/artisan job:preset:cancel <alias>

```

Currently, Octopus has the following job presets for OTP-Bank project:

\* \*\*pg\_update\_services\*\* — updates payment service list every five minutes

\* \*\*notifications\_update\_dispatch\*\* — dispatches notification update jobs every day at 01:00; not in use

To create a new job preset make an extension of the `\App\JobPresets\AbstractJobPreset` class and then register it in `\App\AppServiceProvider` using `\App\JobPresets\JobPresetRegistry`.

## Cron

``\* \* \* \* \* php /path/to/octopus/src/artisan schedule:run >> /dev/null 2>&1``

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t octopus:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 octopus:latest --name octopus --env-file .env

```

Инициализировать структуру БД.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti butler bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

[Пример](https://gitlab.com/selfteam/project-documents/wikis/Переменные-окружения/RegManager)

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | octopus |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша. Для корректной работы Octopus нужно указать \*\*redis\*\* | string | + | file |

| QUEUE\_CONNECTION | Подключение к очереди. Для корректной работы Octopus нужно указать \*\*redis\*\* | string | + | sync |

| REDIS\_HOST | Хост Redis | string | + | |

| REDIS\_PORT | Порт Redis | string | + | 6379 |

| REDIS\_QUEUE | Имя очереди в Redis | string | - | default |

| REDIS\_PASSWORD | Пароль Redis | string | - | |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | oracle |

| DB\_HOST | Адрес сервера БД | string | + | 192.168.60.93 |

| DB\_PORT | Порт сервера БД | string | + | 1521 |

| DB\_DATABASE | Название базы данных | string | + | OTPDM2 |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей. Если не установлено значение true, нужно однократно выполнить команду `php artisan doctrine:generate:proxies` | boolean | - | false |

| # Remote Components | | | | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| PAYMENTGATE\_HOST | Адрес компонента PaymentGate | string | + | |

| POSTIE\_HOST | Адрес компонента Postie | string | + | |

| REGMANAGER\_HOST | Адрес компонента RegManager | string | + | |

| SALT | Соль | string | + | |

| HORIZON\_PREFIX | Префикс хранилища для данных Horizon | string | - | horizon: |

| DEFAULT\_JOBS\_PRESET\_SCHEDULE | Запускаемые по-умолчанию запланированные задачи. Формат: `job\_preset\_alias,another\_job\_preset\_alias` | string | - | |

| SCHEDULED\_JOBS\_PENDING\_TIMEOUT | Время ожидания исполнения запланированных задач в статусе Pending в минутах | integer | - | 5 |

| DEFAULT\_EVENT\_SUBSCRIPTION | Подписки на события по-умолчанию. Формат: `subscriber\_alias:Event\Class,another\_subscriber\_alias:Another\Event\Class` | string | - | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# PaymentGate

Сервис для работы с платежами, черновиками, шаблона. Включат в себя механизм для работы с платежным шлюзом.

## Artisan

@todo

## CronTab

``\* \* \* \* \* php /var/www/.../src/artisan schedule:run >> /dev/null 2>&1``

## Flow платежной сессии

### Создание платежной сессии

Есть несколько вариантов создания платежной сессии:

```

\App\Managers\Payment\AbstractPaymentProcessor::createPaymentByService

\App\Managers\Payment\AbstractPaymentProcessor::createPaymentByOperation

\App\Managers\Payment\AbstractPaymentProcessor::createPaymentByTemplate

\App\Managers\Payment\AbstractPaymentProcessor::createPaymentByInvoice

```

### Выполнение платежа

Выполнение происходит путем последовательного вызова:

Если многошаг:

`\App\Managers\Payment\AbstractPaymentProcessor::submitPaymentForm`

На последнем шаге:

`\App\Managers\Payment\AbstractPaymentProcessor::checkPaymentForm`

После корректного ввода ОТП:

`\App\Managers\Payment\AbstractPaymentProcessor::confirmPaymentForm`

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t paymentgate:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 paymentgate:latest --name paymentgate --env-file .env

```

Инициализировать структуру БД.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti butler bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | paymentgate |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | oracle |

| DB\_HOST | Адрес сервера БД | string | + | 192.168.60.93 |

| DB\_PORT | Порт сервера БД | string | + | 1521 |

| DB\_DATABASE | Название базы данных | string | + | OTPDM2 |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| PAYMENTGATE\_HOST | Адрес компонента PaymentGate | string | + | |

| POSTIE\_HOST | Адрес компонента Postie | string | + | |

| USERBOX\_HOST | Адрес компонента Userbox | string | + | |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| CURRENCY\_HOST | Адрес компонента Currency | string | + | |

| BANK\_LIMIT\_HOST | Адрес компонента Currency | string | + | |

| BANK\_PRODUCT\_HOST | Адрес компонента BankProduct | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| REGMANAGER\_HOST | Адрес компонента RegManager | string | + | |

| USER\_SETTINGS\_HOST | Адрес компонента UserSettings | string | + | |

| SALT | Соль | string | + | |

| # ЦУП | | | | |

| GCPAY\_DATASOURCE\_LOCAL | Вкл/выкл локального источника данных GCPay | boolean | + | |

| GCPAY\_BASE\_URI | Адрес API GCPay | string | + | http://194.85.126.103:10212 |

| GCPAY\_VERSION | Версия протокола GCPay | string | + | 7 |

| GCPAY\_VERSION\_MINOR | Минорная версия GCPay | string | + | 2018-12-02 |

| GCPAY\_TERM\_ID | ??? | string | + | D000002 |

| GCPAY\_CLIENT\_LANG | Язык | string | + | ru |

| GCPAY\_SHOW\_DEFAULT\_ERROR\_MESSAGE | Вкл/выкл отображение отображение сообщения-заглушки на не известную ошибку | bool | - | true |

| GCPAY\_EXCLUDE\_GROUPS | ID групп, которые нужно скрыть (через запятую) | string | + | |

| GCPAY\_FAKE\_PAYMENT\_EXECUTION | Включить фейковую обработку платежа (без отправки в ЦУП) | bool | - | false |

| GCPAY\_PAYMENT\_SERVICE\_IS\_SUSPEND | Идентификатор ЦУП-услуги для жесткого установки флага isSuspend=true | string | - | |

| GCPAY\_LOG\_REQUEST\_NAMES | кастомный набор методов для логирование запросов ЦУП-а | string | - | ReqCalcFee,ReqCalcForm,ReqCheckPay,ReqAuthDebit |

| # ЦУП Группы услуг | | | |

| GCPAY\_GROUP\_ROOT | Идентификатор ЦУП-группы услуг: основное меню | string | + | |

| GCPAY\_GROUP\_COMMUNICATION | Идентификатор ЦУП-группы услуг: сотовые | string | + | |

| GCPAY\_GROUP\_COMMUNAL | Идентификатор ЦУП-группы услуг: жкх | string | + | |

| GCPAY\_GROUP\_PUBLIC\_SERVICES | Идентификатор ЦУП-группы услуг: госуслуги | string | + | |

| GCPAY\_GROUP\_TV | Идентификатор ЦУП-группы услуг: тв | string | + | |

| GCPAY\_GROUP\_INTERNET | Идентификатор ЦУП-группы услуг: интернет | string | + | |

| GCPAY\_GROUP\_DIGITAL\_MONEY | Идентификатор ЦУП-группы услуг: электронные кошельки | string | + | |

| GCPAY\_GROUP\_KINDERGARTEN | Идентификатор ЦУП-группы услуг: детские сады | string | + | |

| GCPAY\_GROUP\_TRANSFER | Идентификатор ЦУП-группы услуг: переводы | string | + | |

| GCPAY\_GROUP\_PHONE | Идентификатор ЦУП-группы услуг: Телефония | string | + | |

| GCPAY\_GROUP\_SECURITY\_SYSTEM | Идентификатор ЦУП-группы услуг: Охранные системы | string | + | |

| GCPAY\_GROUP\_COMMERCE | Идентификатор ЦУП-группы услуг: Коммерция | string | + | |

| GCPAY\_GROUP\_INSURANCE | Идентификатор ЦУП-группы услуг: Страхование | string | + | |

| GCPAY\_GROUP\_TRAVEL | Идентификатор ЦУП-группы услуг: Транспорт и путешествия | string | + | |

| GCPAY\_GROUP\_OTHER | Идентификатор ЦУП-группы услуг: Прочие | string | + | |

| GCPAY\_GROUP\_ENTERTAINMENT | Идентификатор ЦУП-группы услуг: Развлечения | string | + | |

| GCPAY\_GROUP\_CHARITY | Идентификатор ЦУП-группы услуг: Благотворительность | string | + | |

| GCPAY\_GROUP\_EDUCATION | Идентификатор ЦУП-группы услуг: Обучение | string | + | |

| # ЦУП Услуги | | | |

| GCPAY\_INTERNAL\_TO\_YOURSELF | Идентификатор ЦУП-услуги: Внутрибанковские переводы между счетами клиента c конверсией | string | + | |

| GCPAY\_EXTERNAL\_TO\_YOURSELF | Идентификатор ЦУП-услуги: Перевод себе в другой банк | string | + | |

| GCPAY\_EXTERNAL\_TO\_PERSON | Идентификатор ЦУП-услуги: Перевод на счет в другом банке другому физическому лицу | string | + | |

| GCPAY\_EXTERNAL\_TO\_COMPANY | Идентификатор ЦУП-услуги: Юридическим лицам или ИП | string | + | |

| GCPAY\_INTERNAL\_BY\_PHONE | Идентификатор ЦУП-услуги: По номеру телефона | string | + | |

| GCPAY\_INTERNAL\_BY\_CARD\_NUMBER | Идентификатор ЦУП-услуги: По номеру карты | string | + | |

| GCPAY\_INTERNAL\_BY\_ACCOUNT | Идентификатор ЦУП-услуги: Клиенту Банка по номеру счета | string | + | |

| GCPAY\_BUDGET\_TO\_COMPANY | Идентификатор ЦУП-услуги: Бюджетные организации | string | + | |

| GCPAY\_BUDGET\_TAX | Идентификатор ЦУП-услуги: Налоги и сборы | string | + | |

| GCPAY\_CURRENCY\_TRANSFER | Идентификатор ЦУП-услуги: SWIFT перевод другому | string | + | |

| GCPAY\_CURRENCY\_TRANSFER\_YOURSELF | Идентификатор ЦУП-услуги: SWIFT перевод себе | string | + | |

| GCPAY\_OTHER\_CARD | Идентификатор ЦУП-услуги: МИР/VISA/MasterCard (Maestro) РФ | string | + | |

| GCPAY\_EXTERNAL\_FROM\_CARD | Идентификатор ЦУП-услуги: С карты другого банка | string | + | |

| GCPAY\_FINE\_BY\_UIN | Идентификатор ЦУП-услуги: Оплата штрафов по УИН | string | + | |

| GCPAY\_COMMUNAL\_MSK | Идентификатор ЦУП-услуги: ЖКУ Москва | string | + | |

| # QIWI | | | | |

| QIWI\_API\_URL | Адрес API Qiwi | string | + | https://api-test.qiwi.com/partner/b2b2c-subscriptions-api/ |

| QIWI\_AGENT\_ID | Идентификатор агента | string | + | |

| QIWI\_AUTH\_METHOD | Способ аутентификации в Qiwi. TOKEN — аутентификация с токеном, SSL — аутентификация с сертификатом | string | + | |

| QIWI\_AGENT\_TOKEN | Токен агента (обязательно, если QIWI\_AUTH\_METHOD=TOKEN) | string | - | |

| QIWI\_SSL\_CERT\_RESOURCE\_PATH | Путь к сертификату в директории resources (обязательно, если QIWI\_AUTH\_METHOD=SSL) | string | - | |

| QIWI\_SSL\_KEY\_RESOURCE\_PATH | Путь к приватному ключу в директории resources (обязательно, если QIWI\_AUTH\_METHOD=SSL) | string | - | |

| QIWI\_SSL\_CERT\_PASSWORD | Пароль для сертификата (обязательно, если QIWI\_AUTH\_METHOD=SSL) | string | - | |

| # Рапида | | | | |

| RAPIDA\_BASE\_URI | Адрес API Рапиды | string | + | |

| RAPIDA\_METHOD | Функция рапиды | string | + | |

| # Настройка для файлов | | | | |

| STATIC\_FILES\_HOST | Хост, с которого отдавать статику | string | - | |

| STATIC\_SUB\_DIR\_ALIAS | Alias для микросервиса при отдачи статики | string | - | payment-gate |

| SBP\_AMOUNT\_THRESHOLD | Погор суммы для СБП, после которой будут отправляться дополнительные поля | string | - | 15000 |

| # Настройка для антифрода| | | | |

| ANTI\_FRAUD\_ENABLE | Вкл/Выкл отправку данных в антифрод и анализ его ответа. | boolean | - | false |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t postie:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 postie:latest --name postie --env-file .env

```

Инициализировать структуру БД.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti postie bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

# Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | pgsql |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | 5432 |

| DB\_DATABASE | Название базы данных | string | + | postie |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| SALT | Соль | string | + | |

| SMS\_GATE | Способ отправки | string \*\*STUB\*\*, \*\*PROSTOR\*\*, \*\*BUS\*\* | - | STUB |

| SMS\_PROSTOR\_HOST | Адрес API Prostor SMS | string | + | http://api.prostor-sms.ru/ |

| SMS\_PROSTOR\_LOGIN | Логин Prostor SMS | string | + | |

| SMS\_PROSTOR\_PASSWORD | Пароль Prostor SMS | string | + | |

| PUSH\_GATE | Способ отправки Push уведомлений | string \*\*PUSH\*\*, \*\*BUS\*\* | - | STUB |

| MESSAGE\_TEMPLATE\_AUTO\_PAYMENT\_NOTIFY | (Идентификатор шаблона сообщения) Напоминание об автоплатеже в дату автоплатежа | string | - | 386 |

| MESSAGE\_TEMPLATE\_DEFAULT\_CONFIRM | (Идентификатор шаблона сообщения) Подтверждение любых операции кодом ОТР, не имеющих отдельного шаблона | string | - | 667 |

| MESSAGE\_TEMPLATE\_CONFIRM\_TRANSFER | (Идентификатор шаблона сообщения) Подтверждение перевода | string | - | 666 |

| MESSAGE\_TEMPLATE\_CONFIRM\_PAYMENT | (Идентификатор шаблона сообщения) Подтверждение платежа | string | - | 406 |

| MESSAGE\_TEMPLATE\_CONFIRM\_LOGIN | (Идентификатор шаблона сообщения) Подтверждение входа в ИБ/МБ кодом ОТР | string | - | 408 |

| MESSAGE\_TEMPLATE\_CHANGE\_LOGIN | (Идентификатор шаблона сообщения) При изменении Логина в Профиле клиентом | string | - | 409 |

| MESSAGE\_TEMPLATE\_CHANGE\_PASSWORD | (Идентификатор шаблона сообщения) При изменении пароля в профиле | string | - | 506 |

| MESSAGE\_TEMPLATE\_AUTO\_PAYMENT\_ERROR | (Идентификатор шаблона сообщения) При любой ошибке исполнения автоплатежа | string | - | 412 |

| MESSAGE\_TEMPLATE\_SMS\_INFO\_ENABLED | (Идентификатор шаблона сообщения) При подключении смс-информирование на продукт клиента в профиле | string | - | 414 |

| MESSAGE\_TEMPLATE\_CHANGE\_PHONE\_OLD | (Идентификатор шаблона сообщения) Отправляется на старый номер телефона ДБО после изменения номера в ДБО на иной в Профиле | string | - | 415 |

| MESSAGE\_TEMPLATE\_CHANGE\_PHONE\_NEW | (Идентификатор шаблона сообщения) Отправляется на новый номер телефона ДБО после изменения номера в ДБО в Профиле | string | - | 416 |

| MESSAGE\_TEMPLATE\_CONTRACT\_BLOCK | (Идентификатор шаблона сообщения) Блокировка контракта ДБО | string | - | 450 |

| MESSAGE\_TEMPLATE\_CONTRACT\_UNBLOCK | (Идентификатор шаблона сообщения) Разблокировка контракта ДБО | string | - | 451 |

| MESSAGE\_TEMPLATE\_CHANGE\_EMAIL | (Идентификатор шаблона сообщения) Смена e-mail клиента в дбо | string | - | 452 |

| MESSAGE\_TEMPLATE\_CHANNEL\_ENABLE | (Идентификатор шаблона сообщения) Включение доступа к каналу в профиле | string | - | 449 |

| MESSAGE\_TEMPLATE\_CHANNEL\_DISABLE | (Идентификатор шаблона сообщения) Отключение доступа к каналу в профиле | string | - | 448 |

| MESSAGE\_TEMPLATE\_ACCESS\_RESTORE | (Идентификатор шаблона сообщения) Восстановление доступа - смс для отправки действующего логина от ДБО | string | - | 446 |

| MESSAGE\_TEMPLATE\_SUBSCRIPTION\_FOUND | (Идентификатор шаблона сообщения) Найдена подписка | string | - | 447 |

| MESSAGE\_TEMPLATE\_AUTH\_NOTIFY | (Идентификатор шаблона сообщения) Оповещение о входе в ИБ или МБ (по факту входа) | string | - | 407 |

| MESSAGE\_TEMPLATE\_CHANGE\_PHONE\_OTP | (Идентификатор шаблона сообщения) Подтверждение смены номера телефона | string | - | 686 |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t regmanager:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 regmanager:latest --name regmanager --env-file .env

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

# Переменные окружения

[Пример](https://gitlab.com/selfteam/project-documents/wikis/Переменные-окружения/RegManager)

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_URL | Домен web-интерфейса | string | + | https://otp.dev.amict.ru |

| STATIC\_FILES\_HOST | Домен для раздачи статики | string | - | |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | oracle |

| DB\_HOST | Адрес сервера БД | string | + | 192.168.60.93 |

| DB\_PORT | Порт сервера БД | string | + | 1521 |

| DB\_DATABASE | Название базы данных | string | + | OTPDM2 |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| USERBOX\_HOST | Адрес компонента UserBox | string | + | |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| DIRECTORY\_HOST | Адрес компонента Directory | string | + | |

| BANK\_LIMIT\_HOST | Адрес компонента BankLimit | string | + | |

| PAYMENTGATE\_HOST | Адрес компонента PaymentGate | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| POSTIE\_HOST | Адрес компонента Postie | string | + | |

| BANK\_PRODUCT\_HOST | Адрес компонента BankProduct | string | + | |

| USER\_SETTINGS\_HOST | Адрес компонента UserSettings | string | + | |

| SALT | Соль | string | + | |

| FASTGATE\_ENABLED | Вкл/выкл взаимодействие с FastGate | boolean | + | |

| FASTGATE\_API\_URL | Адрес API FastGate | string | + | |

| FASTGATE\_LOGIN | Логин FastGate | string | + | |

| FASTGATE\_PASSWORD | Пароль FastGate | string | + | |

| # Лента операций | | | | |

| SAVE\_OPERATION\_EXT\_DATA | Вкл/Выкл сохранение раскраски операции в DH | bool | - | false | |

| OTP\_DIRECT\_MIGRATIONS\_ENABLE | Вкл/Выкл импорт данных пользователя из OTPDirect/блокировка в старом ДБО | bool | - | false | |

| OTP\_DIRECT\_BLOCK\_CONTRACT | Вкл/Выкл блокировать контракт после миграции пользователя в старом ДБО | bool | - | false | |

| # Сервисы EQL | | | | |

| EQL\_BASE\_URI | Адрес API EQL | string | + | - |

| EQL\_DATASOURCE\_LOCAL | Локальный источник данных | boolean | + | false |

| # Регистрация и миграция | | | | |

| FORCE\_CREATE\_UPDATE\_CONTRACT | Включить принудительное создание контракта | boolean | - | false |

| SETTING\_STORIES\_ENABLED | Вкл/выкл сторис | boolean | - | false |

| SETTING\_STORIES\_HOST | Хост сторис | string | - | |

| SETTING\_MARKET\_RATING\_ENABLED | Вкл/выкл оценок в маркете | boolean | - | false |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# UserBox

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t userbox:latest --pull

```

Подготовить файл с переменными окружения. Описание и пример переменных окружения [здесь](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 userbox:latest --name currency --env-file .env

```

Инициализировать структуру БД и наполнить данными.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti userbox bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate && php artisan db:seed --class BasicSeeder --force"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

|:----|:---------|:----|:---------------|:----------------------|

| APP\_ENV | Текущее окружение | enum: \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | bool | - | false |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | bool | - | false |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| BANK\_PRODUCT\_HOST | Адрес компонента BankProduct | string | + | |

| FACEBOOK\_API\_URL | Адрес FaceBook API | string | - | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Инструкция по развёртыванию

Собрать образ сервиса

```bash

docker build . -t user-settings:latest --pull

```

Подготовить файл с переменными окружения. Описание переменных окружения [ниже](#переменные-окружения)

Запустить в фоне контейнер с нужными значениями переменных окружения, пробросить наружу порт 80

```bash

docker run -d -p 8080:80 user-settings:latest --name user-settings --env-file .env

```

Инициализировать структуру БД.

БД с соответствующим именем должна быть создана заранее.

```bash

docker exec -ti user-settings bash -c \

"cd /var/www/app/src && php artisan doctrine:migrations:migrate"

```

В случае успешного развёртывания сервис должен отдать "1234" в ответ на пустой GET запрос по адресу `http://service-host:80/v1/`.

## Переменные окружения

| Имя | Описание | Тип | Обязательность | Значение по-умолчанию |

| ------ | ------ | ------ | ------ | ------ |

| APP\_ENV | Текущее окружение | \*\*local\*\*, \*\*test\*\*, \*\*testing\*\*, \*\*production\*\* | - | production |

| APP\_DEBUG | Режим отладки | boolean | - | false |

| APP\_TIMEZONE | Часовой пояс | string | - | UTC |

| APP\_SERVICE\_NAME | Название сервиса | string | + | user-settings |

| APP\_LOCALE | Локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| APP\_FALLBACK\_LOCALE | Резервная локаль | Справочник \*\*en\*\*, \*\*ru\*\* | - | en |

| LOG\_CHANNEL | Канал логирования по-умолчанию. Для логирования в stdout указать dockerlogs2 | string | - | stack |

| CACHE\_DRIVER | Драйвер кеша по-умолчанию | string | - | file |

| QUEUE\_DRIVER | Драйвер очереди по-умолчанию | string | - | sync |

| DB\_CONNECTION | Драйвер подключения к БД | string | + | pgsql |

| DB\_HOST | Адрес сервера БД | string | + | |

| DB\_PORT | Порт сервера БД | string | + | 5432 |

| DB\_DATABASE | Название базы данных | string | + | |

| DB\_USERNAME | Имя пользователя БД | string | + | |

| DB\_PASSWORD | Пароль пользователя БД | string | + | |

| DOCTRINE\_SKIP\_SELECT\_FROM\_LOG | Выкл/вкл логирования SELECT запросов к БД | boolean | - | true |

| DOCTRINE\_PROXY\_AUTOGENERATE | Вкл/выкл автоматической генерации проксирующих классов сущностей | boolean | - | false |

| BUTLER\_HOST | Адрес компонента Butler | string | + | |

| BANK\_PRODUCT\_HOST | Адрес компонента BankProduct | string | + | |

| PAYMENTGATE\_HOST | Адрес компонента PaymentGate | string | + | |

| BUS\_MANAGER\_HOST | Адрес компонента BusManager | string | + | |

| OCTOPUS\_HOST | Адрес компонента Octopus | string | + | |

| USERBOX\_HOST | Адрес компонента Userbox | string | + | |

| SALT | Соль | string | + | |