Приложение № 3
к приказу ООО "Абсолют Технологии"
№ 02-01/27-26 от 29.05.2026

Абсолют Технологии
создаем возможности

# "Платежный шлюз 2.0"
# Инструкция по установке.

г. Москва, 2026 год

# ОГЛАВЛЕНИЕ

1. ПАРАМЕТРЫ ОКРУЖЕНИЯ 3
2. ПАРАМЕТРЫ СБОРКИ 3
2.1. Список модулей 3
2.1.1. Сборка jar (для каждого модуля) 3
2.1.2. Сборка образа (для каждого модуля) 3
2.2. Установка linux (systemd) 3
2.3. Установка docker 4
2.4. Установка в kubernetes 4
2.5. Переменные окружения 6

# 1. ПАРАМЕТРЫ ОКРУЖЕНИЯ

- Linux / Docker / Kubernetes
- Java: 21
- PostgreSQL: 14
- Kafka

# 2. ПАРАМЕТРЫ СБОРКИ

## 2.1. Список модулей

- as-fiscalization
- as-payment
- as-payment-gateway
- as-payment-page-gateway
- as-paytool
- as-sber-integrator
- as-tbank-integrator
- as-vtb-integrator
- as-yookassa-integrator

### 2.1.1. Сборка jar (для каждого модуля)

```
mvn clean package
```

**Пример Dockerfile (Dockerfile уже присутствует в каждом модуле)**

```
FROM eclipse-temurin:21-jre-alpine
WORKDIR /opt/app
COPY target/*.jar ./app.jar
CMD java -jar ./app.jar
```

### 2.1.2. Сборка образа (для каждого модуля)

```
docker build <module_image_name> .
```

## 2.2. Установка linux (systemd)

1. Скомпилировать jar для каждого модуля, разместить в /opt/app/
2. Создать env фаилы с параметрами в виде ключ=значение для каждого из модулей и разместить в /etc/default/<module_name>
3. Создать systemd unit фаил с названием модуля и разместить в /etc/systemd/system/<module_name>.unit

```
[Unit]
Description=<module_name>
After=network.target

[Service]
WorkingDirectory=/opt/app
EnvironmentFile=/etc/default/<module_name>
ExecStart=/usr/bin/java -jar <module_name>.jar
Restart=always
```

```
[Install]
WantedBy=multi-user.target
```

4. Выполнить

```
systemctl daemon-reload
systemctl enable <module_name>
systemctl start <module_name>
```

## 2.3. Установка docker

1. Собрать образы для каждого модуля
2. Создать env фаилы с параметрами в виде ключ=значение для каждого из модулей и разместить в /opt/app/<module_name>
3. Запустить контейнеры каждого из модулей

```
docker run -name <module_name> --env-file /opt/app/<module_name> -d -restart unless-
stopped \ <module_image_name>
```

## 2.4. Установка в kubernetes

1. Собрать образы для каждого из модулей, отправить в регистри
2. Применить манифесты к кластеру для каждого из модулей

```
# только для gateway модулей
kind: Ingress
apiVersion: networking.k8s.io/v1
metadata:
  name: <module_name>
  labels:
    app.kubernetes.io/name: <module_name>
spec:
  rules:
    - host: app.example.com
      http:
        paths:
          - path: /
            pathType: ImplementationSpecific
            backend:
              service:
                name: <module_name>
                port:
                  number: 8080
# для всех модулей
---
kind: Service
apiVersion: v1
metadata:
  name: <module_name>
  labels:
    app.kubernetes.io/name: <module_name>
spec:
  type: ClusterIP
  selector:
    app.kubernetes.io/name: <module_name>
  ports:
    - name: http
      protocol: TCP
```

```
      port: 8080
      targetPort: 8080

---
kind: ConfigMap
apiVersion: v1
metadata:
  name: <module_name>
  labels:
    app.kubernetes.io/name: <module_name>
data:
  APP_ENV_EXAMPLE: example_value
---
kind: Secret
apiVersion: v1
metadata:
  name: <module_name>
  labels:
    app.kubernetes.io/name: <module_name>
stringData:
  APP_SECRET_EXAMPLE: example_secret_value
type: Opaque
---
kind: Deployment
apiVersion: apps/v1
metadata:
  name: <module_name>
  labels:
    app.kubernetes.io/name: <module_name>
spec:
  replicas: 1
  selector:
    matchLabels:
      app.kubernetes.io/name: <module_name>
  template:
    metadata:
      labels:
        app.kubernetes.io/name: <module_name>
    spec:
      volumes:
        - name: tmp
          emptyDir:
            sizeLimit: 1Mi
      containers:
        - name: <module_name>
          image: <module_image_name>

          envFrom:
            - configMapRef:
                name: <module_name>
            - secretRef:
                name: <module_name>
                optional: true

          ports:
            - name: http
              containerPort: 8080
              protocol: TCP
```

```
          volumeMounts:
            - name: tmp
              mountPath: /tmp


          resources:
            limits:
              cpu: '1'
              memory: 768Mi
            requests:
              cpu: 10m
              memory: 640Mi

          readinessProbe:
            httpGet:
              path: /actuator/health/readiness
              port: http
              scheme: HTTP
            initialDelaySeconds: 120
            timeoutSeconds: 3
            periodSeconds: 10
            successThreshold: 1
            failureThreshold: 3

          livenessProbe:
            httpGet:
              path: /actuator/health/liveness
              port: http
              scheme: HTTP
            initialDelaySeconds: 200
            timeoutSeconds: 3
            periodSeconds: 30
            successThreshold: 1
            failureThreshold: 30
```

## 2.5. Переменные окружения

### as-fiscalization

```
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
KIAS_BASE-URL
KIAS_AUTH_URL
ORANGE-DATA-MAPPING_CHECK-CLOSE-TAXATION-SYSTEM
ORANGE-DATA-MAPPING_PAYMENT-METHOD-TYPE
ORANGE-DATA-MAPPING_CHECK-CLOSE-PAYMENTS-TYPE
ORANGE-DATA-MAPPING_TAX
ORANGE-DATA-MAPPING_PAYMENT-SUBJECT-TYPE
ORANGE-DATA-HTTP_BASE-URL
SPRING_DATASOURCE_URL
SCHEDULER_CRON-EXPRESSION_START-REPEAT-MESSAGE
SCHEDULER_CRON-EXPRESSION_START-GET-RECEIPT-STATUS
SCHEDULER_CRON-EXPRESSION_START-GET-KIAS-TOKEN
SPRING_SSL_BUNDLE_PEM_CLIENT_KEYSTORE_PRIVATE-KEY
SPRING_SSL_BUNDLE_PEM_CLIENT_KEYSTORE_CERTIFICATE
SPRING_KAFKA_PROPERTIES_BOOTSTRAP-SERVERS
ORANGE-DATA-MAPPING_KEY
ORANGE-DATA-MAPPING_INN
ORANGE-DATA-MAPPING_URL
```

```
KIAS_AUTH_CLIENT-ID
KIAS_AUTH_CLIENT-SECRET
ORANGE-DATA-HTTP_SIGNATURE-PRIVATE-KEY
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SPRING_SECURITY_API-KEY_TOKEN
```

## as-payment

```
INTEGRATION_HTTP_LOGGING-ENABLED
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
KIAS_BASE-URL
KIAS_AUTH_URL
SPRING_DATASOURCE_URL
SCHEDULER_CRON-EXPRESSION_START-GET-KIAS-TOKEN
SCHEDULER_CRON-EXPRESSION_START-REPEAT-MESSAGE
SCHEDULER_CRON-EXPRESSION_START-EMPTY-CONTACTS-NOTIFICATION
AS-PAYTOOL_BASE-URL
AS-TBANK-INTEGRATOR_BASE-URL
AS-SBER-INTEGRATOR_BASE-URL
AS-RSHB-INTEGRATOR_BASE-URL
AS-YOOKASSA-INTEGRATOR_BASE-URL
AS-VTB-INTEGRATOR_BASE-URL
AS-SBER-INTEGRATOR_BASE-URL
PAYOUT_MERCHANT-ID
PAYMENT_AMOUNT_FOR_CHECK_CLIENT_PASSPORT_DATA
SPRING_KAFKA_PROPERTIES_BOOTSTRAP-SERVERS
SPRING_KAFKA_TOPICS_SEND-FISCALIZATION_NAME
CRYPTOGRAM_ENABLED
MERCHANT_CODE_FOR_CUP
BEAUTIFUL-ERROR_SUPPORT-PHONE
SORM_SEND_KAFKA-ENABLED
SORM_SEND_ELASTIC-ENABLED
SORM_SEND_LOG-ENABLED
SORM_ELASTIC_BASE-URL
SORM_ELASTIC_INDEX
AS-NOTIFICATION_BASE-URL
AS-NOTIFICATION_EMPTY-CONTACTS-PROPS_RECIPIENTS
AS-NOTIFICATION_EMPTY-CONTACTS-PROPS_SUBJECT
AS-NOTIFICATION_EMPTY-CONTACTS-PROPS_TEMPLATE-TYPE
APP_LOG_HEADER-ENABLED
AS-NOTIFICATION_AUTH_API-KEY
AS-PAYTOOL_AUTH_API-KEY
AS-RSHB-INTEGRATOR_AUTH_API-KEY
AS-SBER-INTEGRATOR_AUTH_API-KEY
AS-TBANK-INTEGRATOR_AUTH_API-KEY
AS-VTB-INTEGRATOR_AUTH_API-KEY
AS-YOOKASSA-INTEGRATOR_AUTH_API-KEY
CRYPTOGRAM_SECRET-KEY-32-SYMBOLS
KIAS_AUTH_CLIENT-ID
KIAS_AUTH_CLIENT-SECRET
SORM_ELASTIC_PASSWORD
SORM_ELASTIC_USERNAME
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SPRING_SECURITY_API-KEY_TOKEN
```

## as-payment-gateway

```
LOGGING_LEVEL_ROOT
INTEGRATION_HTTP_LOGGING-ENABLED
INTEGRATION_HTTP_PAYMENT-PROPERTY_BASE-URL
INTEGRATION_HTTP_PAYTOOL-PROPERTY_BASE-URL
INTEGRATION_HTTP_OPERATION-PROPERTY_BASE-URL
INTEGRATION_HTTP_TBANK-INTEGRATOR-PROPERTY_BASE-URL
INTEGRATION_HTTP_SBER-INTEGRATOR-PROPERTY_BASE-URL
INTEGRATION_HTTP_YOOKASSA-INTEGRATOR-PROPERTY_BASE-URL
INTEGRATION_HTTP_RSHB-INTEGRATOR-PROPERTY_BASE-URL
AUTH_TOKEN_CLIENT-KEY-LIST
INTEGRATION_HTTP_OPERATION-PROPERTY_API-KEY
INTEGRATION_HTTP_PAYMENT-PROPERTY_API-KEY
INTEGRATION_HTTP_PAYTOOL-PROPERTY_API-KEY
INTEGRATION_HTTP_YOOKASSA-PROPERTY_API-KEY
INTEGRATION_HTTP_SBER-PROPERTY_API-KEY
INTEGRATION_HTTP_RSHB-INTEGRATOR-PROPERTY_API-KEY
```

## as-payment-page-gateway

```
APP_AS-PAYMENT_BASE-URL
APP_LOG_HEADER-ENABLED
APP_AS-PAYMENT_SECURITY_TOKEN
APP_GATEWAY_SECRETKEY
```

## as-paytool

```
SPRING_DATASOURCE_URL
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SPRING_SECURITY_API-KEY_TOKEN
```

## as-sber-integrator

```
APP_PROPERTY_SBER_DYNAMICCALLBACKURL
AS-PAYMENT_CONFIG_BASE-URL
CERT_CERTIFICATEPATH
SPRING_DATASOURCE_URL
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
SPRING_KAFKA_PROPERTIES_BOOTSTRAP-SERVERS
SBER_CONFIG_BASE-URL
SBER_CONFIG_STUB-URL
SBER_CONFIG_USE_STUB
SCHEDULER_CRON-EXPRESSION_PROCESS_START-GET-STUCK-PAYMENTS
SCHEDULER_CRON-EXPRESSION_ERROR_START-PROCESS-SEND-TO-KAFKA
AS-PAYMENT_CONFIG_AUTH_API-KEY
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SPRING_SECURITY_API-KEY_TOKEN
```

## as-tbank-integrator

```
SPRING_DATASOURCE_URL
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
SPRING_KAFKA_PROPERTIES_BOOTSTRAP-SERVERS
TBANK_BASE-URL
TBANK_NOTIFICATION-URL
SCHEDULER_CRON-EXPRESSION_START-REPEAT-MESSAGE
```

```
SCHEDULER_CRON-EXPRESSION_START-CHECK-SUSPENDED-PAYOUTS-PROCESS
SCHEDULER_CRON-EXPRESSION_START-CHECK-SUSPENDED-PAYMENTS-PROCESS
NOTIFY-CONFIG_RECIPIENTS
NOTIFY-CONFIG_SENDER
NOTIFY-CONFIG_TOPIC
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SPRING_SECURITY_API-KEY_TOKEN
TBANK_PASSWORD
```

## as-vtb-integrator

```
SPRING_DATASOURCE_URL
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
SPRING_KAFKA_PROPERTIES_BOOTSTRAP-SERVERS
APP_PROPERTY_VTB_CONFIG_AUTH-BASE-URL
APP_PROPERTY_VTB_CONFIG_VTB-BASE-URL
VTB_BASE-URL
SCHEDULER_CRON-EXPRESSION_PROCESS_START-GET-STUCK-PAYMENTS
SCHEDULER_CRON-EXPRESSION_ERROR_START-PROCESS-SEND-TO-KAFKA
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SECURITY_API-KEY_TOKEN
```

## as-yookassa-integrator

```
SPRING_DATASOURCE_URL
SPRING_KAFKA_PROPERTIES_BOOTSTRAP-SERVERS
YOOKASSA_BASE-URL
SCHEDULER_CRON-EXPRESSION_PROCESS_START-GET-STUCK-PAYMENTS
SCHEDULER_CRON-EXPRESSION_START-CHECK-SUSPENDED-PAYMENTS-PROCESS
LOGGING_LEVEL_ROOT
LOGGING_LEVEL_ORG_APACHE_KAFKA
SERVER_PORT
SPRING_DATASOURCE_PASSWORD
SPRING_DATASOURCE_USERNAME
SPRING_SECURITY_API-KEY_TOKEN
YOOKASSA_CALLBACK-AUTH_PASSWORD
YOOKASSA_CALLBACK-AUTH_USERNAME
```