

Страницы / ... / Описание компонентов ПО

Интеграционный шлюз (mx-gateway)

Создатель Курнос Андрей Валентинович, отредактировано авг 25, 2023 / 243 views since 13 Jul 2023

Программный комплекс курьерской логистики «СблКомпас». Комплект технической документации
PLAROUT.10104-01 13 01 Интеграционный шлюз (mx-gateway). Описание программы

- 1 Общие сведения
- 2 Функциональное назначение
- 3 Описание логической структуры
- 4 Программный интерфейс, входные и выходные данные

4.1 Методы взаимодействия внешних систем с микросервисом (программный интерфейс OpenAPI 3.0)

4.1.1 Состояние заказа POST /v1/orders/status

4.1.2 Информация об обновлениях мобильного приложения POST /v1/updates

4.1.3 Валидация кода вручения POST /v1/order/code/validate

4.1.4 Повторная отправка кода вручения POST /v1/order/code/resend

4.1.5 Загрузка фото в облачное хранилище POST /v1/order/photos/upload

4.1.6 Проверка состояния сервиса GET /monitoring

4.1.7 Информация о сервисе GET /monitoring/info
- 4.2 Методы взаимодействия микросервиса с внешним ПО Zapstore (программный интерфейс JsonRPC 2.0)

4.2.1 Передача статуса успешного завершения задания RPC route_app.setStatusDelivered

4.2.2 Передача статуса отмены задания RPC route_app.setStatusDeliveryFailed

4.2.3 Валидация кода вручения RPC route_app.checkPackageValidationCode

4.2.4 Повторная отправка кода вручения RPC route_app.reloadPackageValidationCode

5 Приложение А. Транслируемые статусы

Общие сведения

Микросервис «Интеграционный шлюз (mx-gateway)» является компонентом программного комплекса курьерской логистики «СблКомпас». Язык разработки сервиса — Java. Для функционирования сервиса необходима система управления базами данных PostgreSQL.

Функциональное назначение

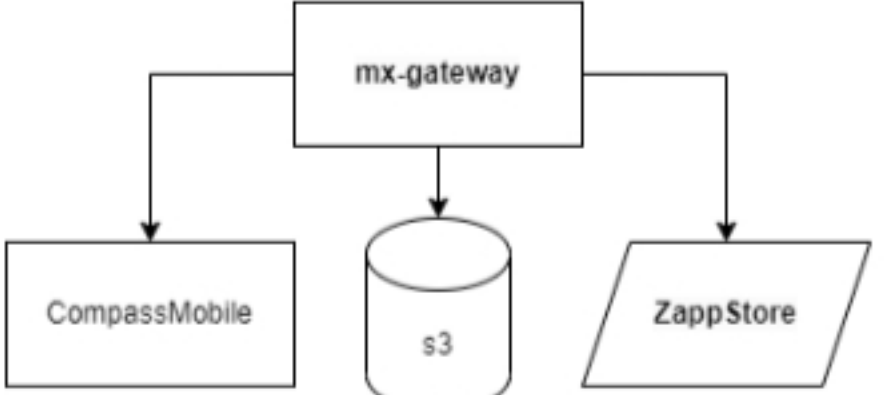
Интеграционный шлюз обеспечивает взаимодействие прикладных компонентов программного комплекса с мобильным приложением (МП) курьера в режиме реального времени. Он реализует механизм отправки статусов выполнения заказов во внешние системы.

Описание логической структуры

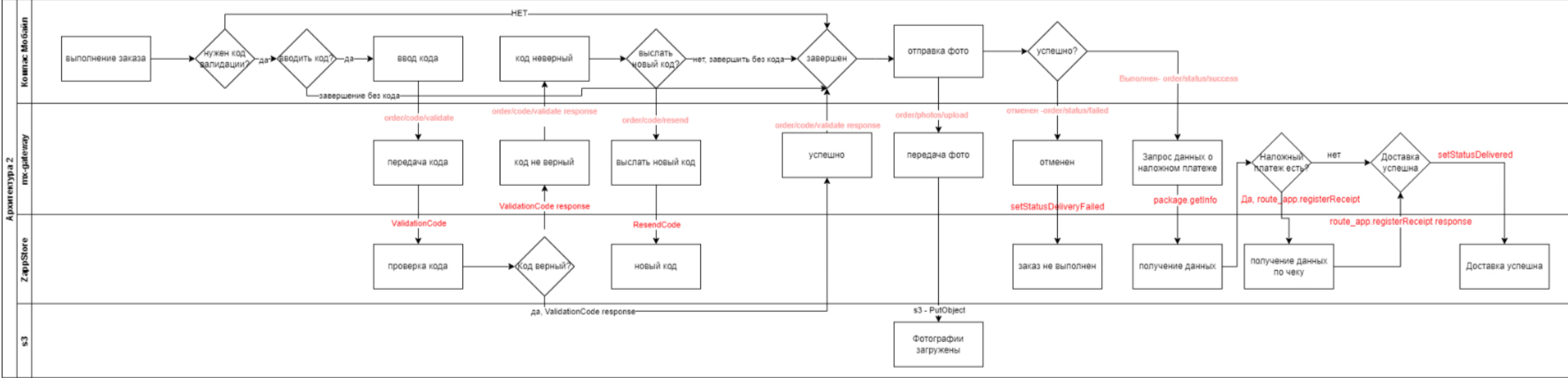
Интеграционный шлюз (mx-gateway) взаимодействует со следующими компонентами программного комплекса и внешними системами:

- CompassMobile — мобильное приложение курьера «КомпасМобайл»;
- ZappStore — внешний комплекс складского ПО;
- s3 — объектное хранилище.

Структурная схема взаимодействия микросервиса с внешними системами:



Функциональная схема взаимодействия микросервиса с внешними системами:



Программный интерфейс, входные и выходные данные

Методы взаимодействия внешних систем с микросервисом (программный интерфейс OpenAPI 3.0)

Сервер (Host): <https://courier-api.sbllogistica.ru>

Состояние заказа

POST /v1/orders/status

Запрос:

header path query body

```
{
  "courierLogin": "string", // Логин курьера
  "orderKind": "string", // Тип заказа
  "orderStatusType": "string", // Тип статуса
  "orderId": "string", // Идентификатор заказа
  "statusDatetime": "2023-07-28T08:55:52.292Z", // Дата и время создания события
  "params": {}, // Фактические параметры заказа в формате json
  "coordinates": {} // Фактические координаты завершения заказа в формате json
}
```

Статусы состояния заказа описаны в Приложении А.

Информация об обновлениях мобильного приложения

POST /v1/updates

Запрос:

header path query body

Authorization: Bearer (jwt-token)
Accept: application/json;charset=UTF-8

Ответ:

201 400 403 404

Ok

Развернуть исходный код

Валидация кода вручения

POST /v1/order/code/validate

Запрос:

header path query body

Authorization: Bearer (jwt-token)
Accept: application/json;charset=UTF-8

Ответ:

200 400 403 404

Ok

Развернуть исходный код

Повторная отправка кода вручения

POST /v1/order/code/resend

Запрос:

header path query body

Authorization: Bearer (jwt-token)
Accept: application/json;charset=UTF-8

Ответ:

201 400 403 404

Created

Загрузка фото в облачное хранилище

POST /v1/order/photos/upload

Запрос:

header path query body

Authorization: Bearer (jwt-token)
Accept: application/json;charset=UTF-8

Ответ:

201 400 403 404

Created

Проверка состояния сервиса

GET /monitoring

Запрос:

header path query body

Authorization: Bearer (jwt-token)
Accept: application/json;charset=UTF-8

Ответ:

200 400 403 404

Ok

Информация о сервисе

GET /monitoring/info

Запрос:

header path query body

Authorization: Bearer (jwt-token)
Accept: application/json;charset=UTF-8

Ответ:

200 400 403 404

Ok

Развернуть исходный код

Методы взаимодействия микросервиса с внешним ПО Zapstore (программный интерфейс JsonRPC 2.0)

POST <https://stock.zapstore.pro/api/jsonrpc?key={token}>

stock — код склада, на котором идет работа с заказом ("msk-2")

token — токен авторизации, выдается администратором для выбранной среды (prod, test) и используется во всех запросах для любого склада ("dc599788-8419-4721-bafa-aca1c76b361")

Передача статуса успешного завершения задания

RPC route_app.setStatusDelivered

Запрос

Ответ

Свернуть исходный код

```
{
  "id": "uuid", // уникальный идентификатор запроса ("9a049144-0759-4497-8a00-e9369896b43e")
  "jsonrpc": "2.0",
  "method": "route_app.setStatusDelivered",
  "params": [
    $userBadge, // идентификатор курьера Zapstore ("driver.Ivanov")
    $barcode, // штрихкод посылки (номер задания) ("RP99930232")
  ]
}
```

Транслируемые статусы описаны в Приложении А.

Передача статуса отмены задания

RPC route_app.setStatusDeliveryFailed

Запрос

Ответ

Свернуть исходный код

```
{
  "id": "uuid", // уникальный идентификатор запроса ("9a049144-0759-4497-8a00-e9369896b43e")
  "jsonrpc": "2.0",
  "method": "route_app.setStatusDeliveryFailed",
  "params": [
    $userBadge, // идентификатор курьера Zapstore ("driver.Ivanov")
    $barcode, // штрихкод посылки (номер задания) ("RP99930232")
    $reason, // текст причины отмены ("Нет связи с адресатом")
    $comment // комментарий ("получатель не хочет брать трубку")
  ]
}
```

Транслируемые статусы описаны в Приложении А.

Валидация кода вручения

RPC route_app.checkPackageValidationCode

Запрос

Ответ

Свернуть исходный код

```
{
  "id": "uuid", // уникальный идентификатор запроса ("9a049144-0759-4497-8a00-e9369896b43e")
  "jsonrpc": "2.0",
  "method": "route_app.checkPackageValidationCode",
  "params": [
    $packageCode, // номер заказа ("RP99930232")
    $validationCode // код валидации ("44584")
  ]
}
```

Повторная отправка кода вручения

RPC route_app.reloadPackageValidationCode

Запрос

Ответ

Свернуть исходный код

```
{
  "id": "uuid", // уникальный идентификатор запроса ("9a049144-0759-4497-8a00-e9369896b43e")
  "jsonrpc": "2.0",
  "method": "route_app.reloadPackageValidationCode",
  "params": [
    $packageCode // номер заказа ("RP99930232")
  ]
}
```

Приложение А. Транслируемые статусы

Микросервис mx-gateway транслирует состояние выполнения задания курьером из мобильного приложения в систему Zapstore. Реализована передача следующих статусов:

Статус	Описание	Обстоятельства формирования статуса	Причины формирования статуса
«Забрано»	Статус успешного выполнения задания	Статус устанавливается курьером в момент получения им отправления для дальнейшей транспортировки по маршруту	—
«Отмена»	Статус неуспешного выполнения задания	Статус устанавливается курьером в момент отказа клиента от сбора (доставки). Статус формируется только после согласования с логистом	<div>«Нет связи с адресатом»<div>Связь с клиентом отсутствует за час до прибытия курьера или в течение 15 минут после прибытия курьера по адресу</div></div> <div>«Отказ по телефону»<div>Курьер связался с клиентом, который отказывается от сбора (доставки) отправления</div></div> <div>«Отказ на месте»<div>Курьер прибыл к клиенту, который отказывается от сбора (доставки) отправления</div></div> <div>«Перенос даты по просьбе адресата»<div>Клиент не может получить (отправить) посылку в назначенную дату и инициирует перенос даты прибытия курьера</div></div> <div>«Согласование смены адреса и переноса доставки»<div>Клиент не может получить (отправить) посылку по указанному адресу и инициирует перенос адреса прибытия курьера</div></div> <div>«Технические причины»<div>любая другая причина</div></div>