

[Решаемая задача](#)

[Системы \(гlossарий\)](#)

[Функциональные требования](#)

[Нефункциональные требования](#)

[Общее описание интеграции систем](#)

[Диаграмма потоков данных](#)

[Применяемые технологии](#)

[Общие требования к журналированию событий](#)

[Обработка и протоколирование ошибок](#)

[Применяемые технологии](#)

[Модель данных](#)

[Сценарии интеграции](#)

[Передача и преобразование данных](#)

[Обновление данных в CRM-системе](#)

[Обновление данных учетной записи пользователя](#)

[Преобразование данных для поля Пол](#)

[Заведение карточки товара](#)

[Регистрационные данные продавца](#)

[Запрос с сайта в систему учета товара информации о товаре](#)

[Получение данных о товаре для отображения на сайте](#)

[Описание технологии передачи данных](#)

[Создание товара](#)

Решаемая задача

На курсе мы работаем с проектом «Универсальный интернет-магазин».

Предварительно был проведён бизнес анализ и получены следующие результаты.

Проблема

За предыдущий год компания потеряла 20% прибыли из-за отсутствия логистики.

Каждый заказ отвозили отдельно. Оплата труда сотрудников доставки выше, чем

могла бы быть с логистикой. Плохие отзывы клиентов, так как конкуренты могут

предложить доставку с погрешностью в час, в нашем случае погрешность составляет 1 рабочий день.

Цели для бизнеса

Увеличить прибыль на 25% в течение года за счет предоставления более точных и сжатых сроков доставки, повышающих конкурентоспособность компании на рынке, а также лояльность клиентов.

Как конкретно бизнес придёт к этой цели

Система позволит:

- Снизить затраты на амортизацию транспортных средств на 15%.
- Уменьшить расходы на оплату труда штату водителей транспортных средств на 20%.
- Снизить расходы на горючее на 20% в течение трёх месяцев после начала работы системы за счёт оптимизации маршрутов, соблюдения очередности загрузки товаром транспортного средства в соответствии с очередностью пунктов доставки.
- Увеличить вместимость склада на 10% за счет оптимизации загрузки склада.

Задача проекта

Автоматизировать логистический процесс начиная от получения заказа в систему и заканчивая непосредственной доставкой, за счёт проектирования межсистемной интеграции.

Системы (гlossарий)

Система	Значение	Функциональность	Пример
PIM	Product Information Management Система управления данными о товарах	Агрегация и хранение информации о товарах. Обеспечение единообразия данных за счет автоматического экспорта данных в другие системы.	PIMCore, Akeneo

WMS	Warehouse Management System Система управления складом	Автоматизация управления бизнес-процессами складской работы предприятия.	Axelot, 1C WMS
TMS	Transportation management system Система управления грузоперевозками	Позволяет поставщикам, перевозчикам и заказчикам автоматизировать логистические процессы, сокращать расходы на перевозки и экономить время.	Axelot, 1C TMS
OMS	Order Management System Система управления заказами	Преобразует коммерческий заказ клиента в логистические заявки по видам потоков товаров и просчитывает перевозку по каждому отрезку цепи поставок с учетом нормативных сроков	Starfish
CM	Система маршрутизации	Распределение заказов по исполнителям и построение для них оптимальных маршрутов	Яндекс-маршрутизация, Atlas

Функциональные требования

Код требования	Требование
Маршрутизация	
Маршрутизация. Формирование маршрута	Система маршрутизации должна сформировать маршрут доставки

Маршрутизация. Передача сформированног о маршрута	Система маршрутизации должна передавать сформированный маршрут в модуль TMS
Маршрутизация. Передача статуса заказа	Система маршрутизации должна передавать статус заказа в систему OMS
Маршрутизация. ПолучениеТС	Система маршрутизации должна запросить данные о ТС и графиках работы курьеров из модуля TMS
Маршрутизация. ПолучениеЗаказо в	Система маршрутизации должна запросить данные о заказах из системы OMS
Маршрутизация. ПолучениеГабари товТоваров	Система маршрутизации должна запросить данные о габаритах товаров из модуля PIM
OMS	
OMS.ОбработкаЗа каза	Система OMS должна обработать заказ и присвоить ему статус
OMS. ПолучениеОстатк ов	Система OMS должна запросить данные об остатках товаров на складе из модуля WMS
WMS	
WMS. Передача. ВоротаСклада	Система WMS должна передать в TMS номер ворот склада и время погрузки
WMS.Передача статуса заказа	Система WMS должна передать в OMS статус заказа
PIM	
PIM. ПередачаТовара. WMS	PIM должен передавать информацию о товарах в WMS

PIM.ПередачаТовара.OMS	PIM должен передавать информацию о товарах в OMS
TMS	
TMS.Передача.ЗадачаНаСборку	TMS должен передать в систему WMS задачу на сборку
TMS.СтатусДоставки	TMS должен передать в систему OMS уведомление об изменениях в статусе и графике доставки

Нефункциональные требования

Код требования	Требование
<i>TMS.Произв</i>	<i>Соответствие рекомендациям в части производительности к Custom Enterprise Desktop App - 2 класс</i>
TMS.Произв.КоличествоПользователей	Система TMS должна позволять одновременную работу до 30 человек (включительно)
TMS.Произв.Запросы.Минута	Система TMS должна позволять обрабатывать до 10 запросов (включительно) в минуту
TMS.Произв.Транзакции.Минута	Система TMS должна позволять обрабатывать до 300 транзакций (включительно) в минуту
TMS.Произв.Транзакции.Длительность	В системе TMS длительность типовой транзакции в 80% случаев должна составлять не более 3 секунд, в 95% не более 5 секунд
<i>WMS.Произв</i>	<i>Соответствие рекомендациям в части производительности к Custom Enterprise Desktop App - 2 класс</i>
WMS.Произв.КоличествоПользователей	Система WMS должна позволять одновременную работу не менее 70 человек
WMS.Произв.Запросы.Минута	Система WMS должна позволять обрабатывать до 30 запросов (включительно) в минуту
WMS.Произв.Транзакции.Секунда	Система WMS должна позволять обрабатывать не менее 50 транзакций в секунду

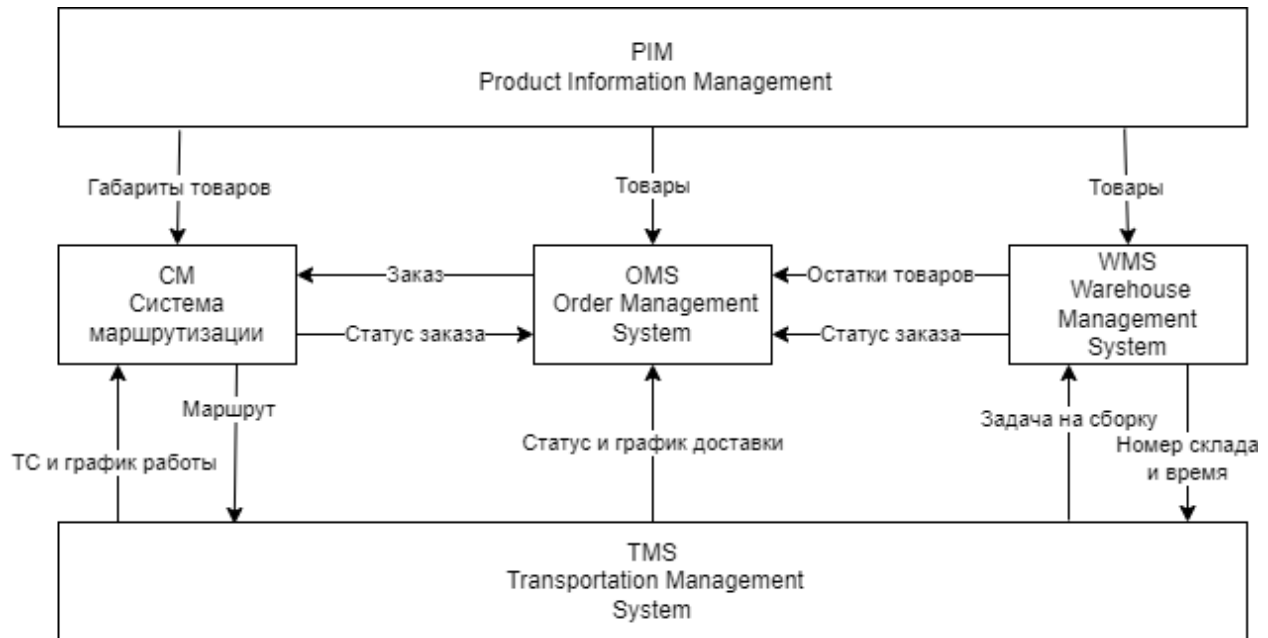
WMS.Произв.Транзакции.День	Система должно позволять обрабатывать не менее 1 800 000 транзакций в день.
WMS.Произв.Транзакции.Длительность	В системе WMS длительность типовой транзакции в 80% случаев должна составлять не более 3 секунд, в 95% не более 5 секунд
<i>OMS.Произв</i>	<i>Соответствие рекомендациям в части производительности к B2B SaaS - 3 класс</i>
OMS.Произв.КоличествоПользователей	Система OMS должна позволять одновременную работу до 50 человек (включительно)
OMS.Произв.Запросы.Минута	Система OMS должна позволять обрабатывать до 10 запросов (включительно) в минуту
OMS.Произв.Транзакции.Секунда	Система OMS должна позволять обрабатывать не менее 50 транзакций в секунду
OMS.Произв.Транзакции.Длительность	В системе OMS длительность типовой транзакции в 80% случаев должна составлять не более 3 секунд, в 95% не более 5 секунд
<i>PIM.Произв</i>	<i>Соответствие рекомендациям в части производительности к B2B SaaS - 3 класс</i>
<i>Маршрутизация.Произв</i>	<i>Соответствие рекомендациям в части производительности к B2B SaaS - 3 класс</i>
<i>PIM.Надеж</i>	<i>Спец. требование - на 1 ступень ниже рекомендаций в части надёжности к B2B SaaS, т.е. 3 минус 1 = 2 класс</i>

PIM.Надеж.Восстановление	В системе PIM длительность восстановления после сбоя не более 30 секунд
<i>OMS.Надеж</i>	<i>Соответствие рекомендациям в части надёжности к B2B SaaS - 3 класс</i>
OMS.Надеж.Сбой	В системе OMS вероятность сбоя (без ущерба для безопасности) не более 0.1%
<i>Маршрутизация.Надеж</i>	<i>Соответствие рекомендациям в части надёжности к B2B SaaS - 3 класс</i>
Маршрутизация.Надеж.Восстановление	В системе маршрутизации длительность восстановления после сбоя не более 5 секунд
<i>TMS.Надеж</i>	<i>Соответствие рекомендациям в части надёжности к Custom Enterprise Desktop App - 2 класс</i>
TMS.Надеж.Сбой	В системе TMS вероятность сбоя (без ущерба для безопасности) не более 1%
<i>WMS.Надеж</i>	<i>Соответствие рекомендациям в части надёжности к Custom Enterprise Desktop App - 2 класс</i>
WMS.Надеж.Сбой	В системе WMS вероятность сбоя (без ущерба для безопасности) не более 1%
<i>PIM.Дост</i>	<i>Соответствие рекомендациям в части доступности к B2B SaaS - 3 класс</i>

PIM.Дост.Простой.День	Система PIM должна обеспечивать уровень доступности, при котором суммарное время простоя системы не превышает 15 секунд в рабочий день с 6 утра до 2 часов ночи (20 часов) по московскому времени
OMS.Дост	Соответствие рекомендациям в части доступности к B2B SaaS - 3 класс
OMS.Дост.Простой.День	Система OMS должна обеспечивать уровень доступности, при котором суммарное время простоя системы не превышает 15 секунд в рабочий день с 6 утра до 2 часов ночи (20 часов) по московскому времени
Маршрутизация.Дост	Соответствие рекомендациям в части доступности к B2B SaaS - 3 класс
Маршрутизация.Дост.Месяц	Система маршрутизации должна обеспечивать уровень доступности, при котором суммарное время простоя системы не превышает 1 минуту в месяц
TMS.Дост	Соответствие рекомендациям в части доступности к Custom Enterprise Desktop App - 1 класс
TMS.Дост.Простой.Час	Система TMS должна обеспечивать уровень доступности, при котором суммарное время простоя системы не превышает 3 минуты в час
WMS.Дост	Соответствие рекомендациям в части доступности к Custom Enterprise Desktop App - 1 класс
WMS.Дост.Простой.День	Система WMS должна обеспечивать уровень доступности, при котором суммарное время простоя системы не превышает 30 минут в рабочий день с 9 до 18 часов по московскому времени

Общее описание интеграции систем

Диаграмма потоков данных



Регламент взаимодействия систем

№	Система-источник	Система-приемник	Передаваемые данные	Полный объем/изменения	Периодичность обмена/событие
1.	WMS	OMS	Остатки товаров	Полный объем	По событию (по запросу)
2	OMS	Система маршрутизации	Заказ	Полный объем	По событию
3	PIM	Система маршрутизации	Габариты товара	Полный объем	По событию (по запросу)
4	TMS	Система маршрутизации	ТС и график работы	Полный объем	По расписанию. Раз в неделю по пятницам в 15:00
5	Система маршрутизации	TMS	Сформированный маршрут	Изменения	По событию (по готовности)
6	Система маршрутизации	OMS	Статус заказа	Изменения	По событию (по готовности маршрута)
7	TMS	WMS	Задача на сборку	Полный объем	По событию
8	WMS	TMS	Номер склада и время	Полный объем	По событию
9	TMS	OMS	Статус и график доставки	Изменения	По событию
10	PIM	WMS	Товары	Изменения	По расписанию. Раз в сутки в 02:00 ночи

Общие требования к журналированию событий

Код требования	Требование
Журналирование.Формат отчета	<p>Отчёт в табличном виде должен содержать следующие столбцы с возможностью фильтрации по столбцам:</p> <ul style="list-style-type: none"> • тип события • id события • дата/время события • описание данных-объекта передачи • система-создатель события • система-отправитель данных • система-получатель данных <p>По умолчанию - отчёт выгружает данные с 00:00 1-го числа текущего месяца по текущее время</p>
Журналирование.Отчет. Режимы формирования	<p>Отчёт должен формироваться двумя путями:</p> <ul style="list-style-type: none"> • по расписанию (ежедневно в 06:00) • по запросу
Журналирование.Период хранения записей	Логи должны храниться в течение 30 календарных дней
Журналирование. Архивирование логов	Логи должны архивироваться через 30 календарных дней, а через 3 года - удаляться из архива

Обработка и протоколирование ошибок

Код требования	Требование
Ошибки.Лог	Ошибки должны журналироваться в лог ошибок, отдельный от других типов событий. При этом форматы и

	<p>периодичность отчётов, а также сроки хранения и архивирования - должны быть идентичны параметрам, указанным для лога событий.</p>
<p>Ошибки.Уведомление администратору</p>	<p>Все системы (PIM, CM, OMS, WMS, TMS) должны иметь возможность отображать сообщения о текущих ошибках администратору системы в режиме реального времени</p> <ul style="list-style-type: none"> • Уведомление в логе журнала интеграции с временем ошибки • Уведомление по e-mail <p>Формат сообщений - в соответствии с требованием Журналирование.Формат Отчета</p>
<p>ОшибкиБизнесПравил. Уведомление Пользователю</p>	<p>Все системы (PIM, CM, OMS, WMS, TMS) должны отображать сообщение об ошибке бизнес-правил пользователю</p>

Применяемые технологии

Технологии:

- HTTPS/REST удалённый вызов процедуры
- HTTPS/XML передача файлов
- Общая БД
- Обмен сообщениями

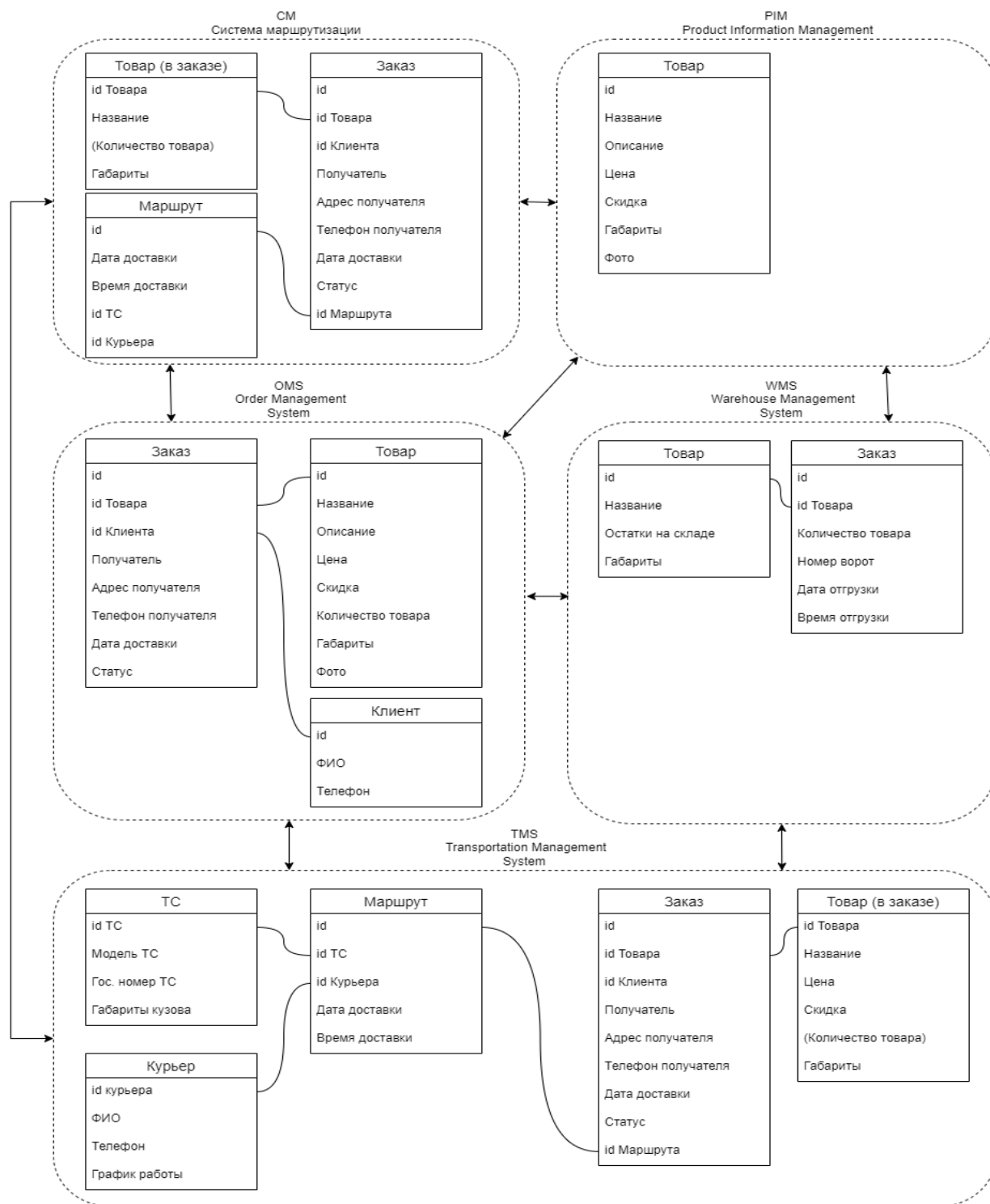
Система-источник	Система-получатель	Данные	Технология
PIM	Система маршрутизации	Габариты товаров	HTTPS/REST удалённый вызов процедуры
PIM	OMS	Товары	HTTPS/REST удалённый вызов процедуры
PIM	WMS	Остатки товаров	Общая БД
OMS	Система маршрутизации	Заказ	HTTPS/XML передача файлов
TMS	Система маршрутизации	ТС и график работы	HTTPS/XML передача файлов
Система маршрутизации	TMS	Маршрут	HTTPS/XML передача файлов
TMS	WMS	Задача на сборку	HTTPS/XML передача файлов

WMS	TMS	Номер склада и время	Обмен сообщениями
Система маршрутизации	OMS	Статус заказа	Обмен сообщениями
TMS	OMS	Статус заказа	Обмен сообщениями
WMS	OMS	Статус заказа	Обмен сообщениями

Инструменты тестирования и документирования решения:

- PostMan
- Swagger

Модель данных



База данных

The screenshot shows the Hasura Data Manager interface. On the left, the 'Data Manager' sidebar lists databases and tables. The main area displays the 'good_order' table with tabs for 'Browse Rows', 'Insert Row', 'Modify', 'Relationships', and 'Permissions'. The 'Relationships' tab is active, showing a table with columns: NAME, SOURCE, TYPE, and RELATIONSHIP. It lists two relationships: 'good' and 'order', both of type 'Array'. The 'good' relationship connects 'good_order / good_id' to 'good / id'. The 'order' relationship connects 'good_order / order_id' to 'order / id'. Each relationship has 'Rename' and 'Remove' buttons. At the bottom, there is an 'Add Relationship' button and a legend for symbols used in the relationship definitions.

NAME	SOURCE	TYPE	RELATIONSHIP
good	default	Array	good_order / good_id → good / id
order	default	Array	good_order / order_id → order / id

Пример запроса GraphQL - Query

The screenshot shows a GraphQL IDE interface. The left pane contains a query named 'QueryExample1' with the following structure:

```
1 query QueryExample1 {
2   good {
3     name
4     id
5   }
6   order {
7     receiver
8     id
9   }
10 }
11
```

Below the query is a section for 'QUERY VARIABLES' with one variable:

```
1
```

The right pane shows the JSON response:

```
{
  "data": {
    "good": [
      {
        "name": "Стул",
        "id": 1
      },
      {
        "name": "Стол",
        "id": 2
      },
      {
        "name": "Ноутбук",
        "id": 3
      }
    ],
    "order": [
      {
        "receiver": "Филимонов А.А.",
        "id": 1
      }
    ]
  }
}
```

At the bottom, the response details are shown: RESPONSE TIME 515 ms, RESPONSE SIZE 204 bytes.

Пример запроса GraphQL - Mutation

The screenshot shows a GraphQL IDE interface. On the left, a mutation query is written: `mutation MyMutationExample2 { insert_good_order(objects:[{ good_id:2, order_id:3 },{ good_id:1, order_id:3 }]) { returning { id } } }`. On the right, the JSON response is displayed: `{ "data": { "insert_good_order": { "returning": [{ "id": 7 }, { "id": 8 }] } } }`. The interface includes buttons for Prettify, History, Explorer, Cache, Code Exporter, REST, Derive action, and Analyze.

Сценарии интеграции

ID: UC-1.1
Название: OMS получает из WMS остатки товара по заказу
Участники: OMS, WMS, Колл-центр
Предусловия: Авторизованный пользователь OMS формирует заказ
Триггер: В OMS поступил новый заказ
Постусловия: В OMS по каждому товару в заказе известен остаток

Основной поток:

1. OMS отправляет в WMS перечень id товаров в заказе.
2. WMS возвращает в OMS информацию об остатках:
 - id товара
 - остатки товара
3. OMS получает и валидирует информацию. Проверяется выполнение для каждого id товара следующего условия: 'Остатки товара' \geq 'Количество в заказе'
3. Все товары в достаточном количестве
4. OMS изменяет статус заказа на "Подтвержден"
5. Сценарий завершён

Расширение:

- За. Товаров недостаточно для заказа
- За.1. OMS изменяет статус заказа на "Требуется корректировка"
- За.2. OMS добавляет новую заметку в заказ по каждому недостающему товару по шаблону "Недостаточно товарных остатков. <Название товара>. На складе - <остатки товара>, в заказе - <количество в заказе>. Требуется корректировка позиций заказа с клиентом"
- За.3. OMS уведомляет Колл-центр о новой заметке в заказе
- За.4. Сценарий завершён

ID: UC-1.2

Название: OMS передаёт новый заказ в Систему маршрутизации

Участники: OMS, Система маршрутизации

Предусловия: В Системе маршрутизации есть данные от ТС и графике работы курьеров (получено из TMS)

Постусловия: В СМ есть данные о заказе и маршруте, в ОМС статус заказа - "Маршрут сформирован"

Основной поток:

1. ОМС передаёт в СМ новый заказ:
 - id заказа
 - id товаров (и количество товаров с этим id в заказе)
 - id клиента
 - Получатель
 - Адрес получателя
 - Телефон получателя
 - Дата доставки
 - Статус
2. СМ получает и валидирует данные заказа. Проверяются условия:
(А) соответствие номера телефона получателя маске "+7 xxx xx xx xxx",
(В) Длина фио получателя > 0,
(С) id заказа, (D) id клиента, (Е) id товаров - непустые/ненулевые,
(F) Количество товаров > 0,
(G) дата доставки - не раньше следующего дня от текущей даты,
(Н) Статус = "Подтвержден" или "Оплачен".
3. Валидация успешно пройдена.
4. Сценарий завершён

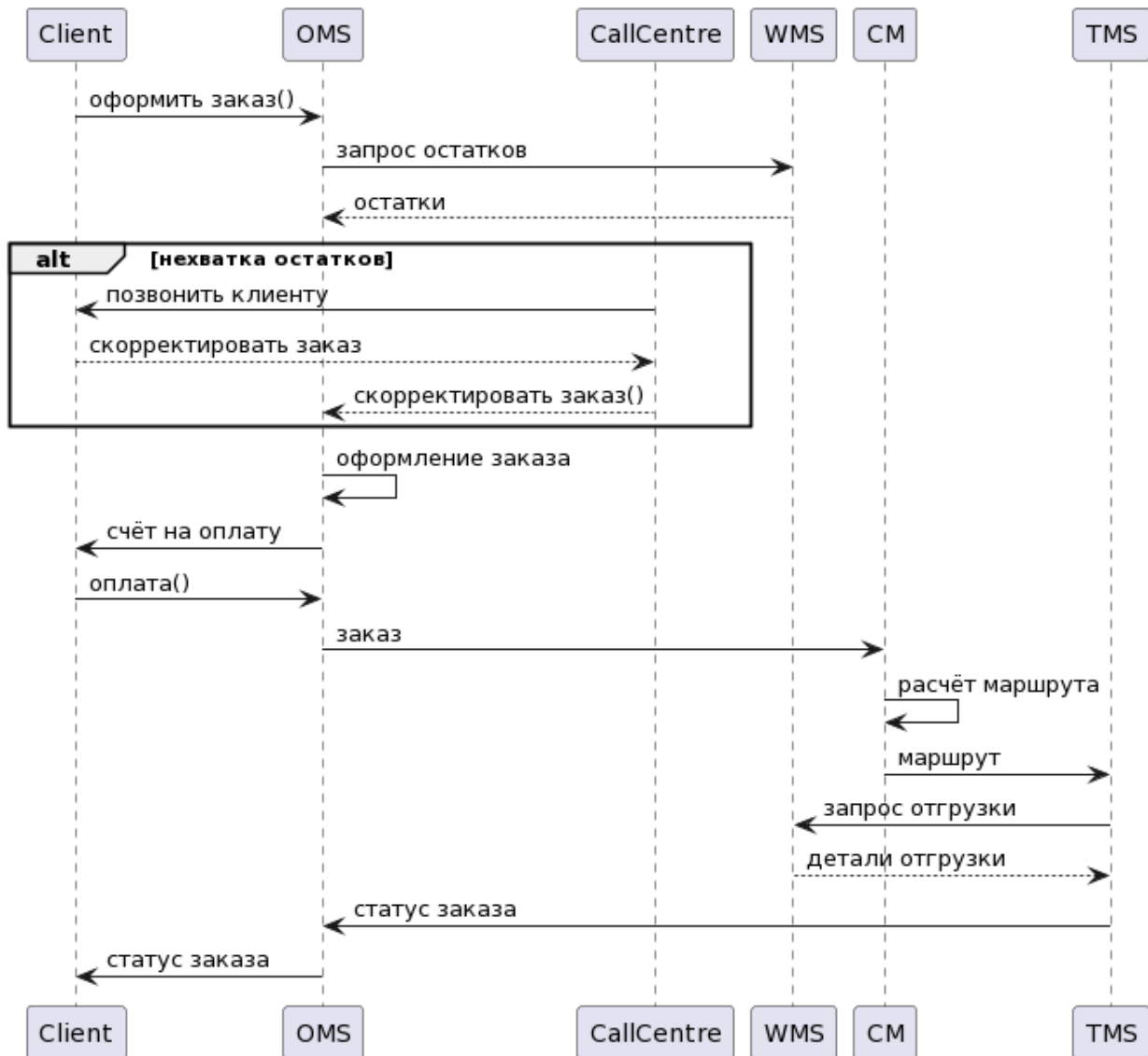
Расширение:

За. Валидация не пройдена

За.1. СМ возвращает в ОМС id заказа и статус заказа "Ошибка в атрибутах заказа. XYZ", где под XYZ указываются нарушенные условия пункта 2 - от А до Н

За.2. Сценарий завершён

Диаграмма последовательности



Передача и преобразование данных

Передача заказа

Система-источник				Система-приемник			
OMS				Система маршрутизации			
Объект/ атрибут		Тип, значение	Обязательность (+/-)	Объект/ атрибут		Тип, значение	Обязательность (+/-)
Order	Заказ	Object	+	Order	Заказ	Object	+
id	id заказа	string	+	id	id заказа	string	+
id_client	id клиента	string	+	id_client	id клиента	string	+
receiver	Получатель	string	-	receiver	Получатель	string	-

Учебный проект проектирования интеграций ИТ-систем

address	Адрес получател я	data	+	address	Адрес получателя	data	+
phone	Телефон получател я	string	+	phone	Телефон получателя	string	+
status	Статус заказа	string	+	status	Статус заказа	string	+
delivery_date	Дата доставки	date	+	delivery_date	Дата доставки	date	+
Goods	Товары в заказе	Object	+	Goods	Товары в заказе	Object	+
name	Название	string	+	name	Название	string	+
id_order	id заказа	string	+	id_order	id заказа	string	+
id_good	id товара	string	+	id_good	id товара	string	+

q_good	Количество о товара	numeric	+	q_good	Количество товара	numeric	+
--------	------------------------	---------	---	--------	----------------------	---------	---

Обновление статуса заказа

Система-источник				Система-приемник			
Система маршрутизации				OMS			
Объект/ атрибут		Тип, значен ие	Обязате льность (+/-)	Объект/ атрибут		Тип, значение	Обязатель ность (+/-)
Order	Заказ	Object	+	Order	Заказ	Object	+
status	Статус заказа	string	+	status	Статус заказа	string	+

Описание технологии передачи данных

Пример сообщения, передаваемого из OMS в Систему маршрутизации [XML]

<https://drive.google.com/file/d/14fsOsSGcO33qBvZB4yxWUiOhNuetu2MI/view?usp=sharing>

```
<?xml version="1.0" encoding="UTF-8"?>
<root>

<__EQ__schema>https://json-schema.org/draft/2019-09/schema</__EQ__schema>
  <__EQ__id>http://example.com/example.json</__EQ__id>
  <type>object</type>
  <default></default>
  <title>Root Schema</title>
  <required>id</required>
  <required>id_client</required>
  <required>receiver</required>
  <required>address</required>
  <required>phone</required>
  <required>status</required>
  <required>delivery_date</required>
  <required>goodsInOrder</required>
  <properties>
    <id>
      <type>string</type>
      <default string="true"/>
      <title>The id Schema</title>
      <examples array="true">order000001</examples>
    </id>
    <id_client>
      <type>string</type>
      <default string="true"/>
      <title>The id_client Schema</title>
      <examples array="true">client010</examples>
    </id_client>
    <receiver>
      <type>string</type>
      <default string="true"/>
      <title>The receiver Schema</title>
      <examples array="true">Устинов Н.А.</examples>
    </receiver>
    <address>
      <type>string</type>
      <default string="true"/>
      <title>The address Schema</title>
      <examples array="true">Москва, Кутузовский пр.,32А</examples>
    </address>
    <phone>
      <type>string</type>
      <default string="true"/>
      <title>The phone Schema</title>
      <examples array="true">+79162820282</examples>
```

```

</phone>
<status>
  <type>string</type>
  <default string="true"/>
  <title>The status Schema</title>
  <examples array="true">Подтверждён</examples>
</status>
<delivery_date>
  <type>string</type>
  <default string="true"/>
  <title>The delivery_date Schema</title>
  <examples array="true">31.07.2023</examples>
</delivery_date>
<goodsInOrder>
  <type>array</type>
  <default empty-array="true"></default>
  <title>The goodsInOrder Schema</title>
  <items>
    <type>object</type>
    <title>A Schema</title>
    <required>id_good</required>
    <required>name</required>
    <required>q_good</required>
    <properties>
      <id_good>
        <type>string</type>
        <title>The id_good Schema</title>
        <examples>goods0002</examples>
        <examples>goods0004</examples>
      </id_good>
      <name>
        <type>string</type>
        <title>The name Schema</title>
        <examples>Мыло душистое</examples>
        <examples>Полотенце пушистое</examples>
      </name>
      <q_good>
        <type>integer</type>
        <title>The q_good Schema</title>
        <examples number="true">1</examples>
        <examples number="true">2</examples>
      </q_good>
    </properties>
    <examples>
      <id_good>goods0002</id_good>
      <name>Мыло душистое</name>
      <q_good number="true">1</q_good>
    </examples>
    <examples>
      <id_good>goods0004</id_good>
      <name>Полотенце пушистое</name>
      <q_good number="true">2</q_good>
    </examples>
  </items>
</goodsInOrder>

```

```

</items>
<examples array="true">
  <examples>
    <id_good>goods0002</id_good>
    <name>Мыло душистое</name>
    <q_good number="true">1</q_good>
  </examples>
  <examples>
    <id_good>goods0004</id_good>
    <name>Полотенце пушистое</name>
    <q_good number="true">2</q_good>
  </examples>
</examples>
</goodsInOrder>
</properties>
</root>

```

Описание XML-схемы для сообщения, передаваемого из OMS в Систему маршрутизации [XSD]

https://drive.google.com/file/d/1SANyLRqFRZOI7htUsxTyAZ_8-HJkSvP6/view?usp=sharing

```

<?xml version="1.0" encoding="utf-8"?>
<!-- Created with Liquid Technologies Online Tools 1.0
(https://www.liquid-technologies.com) -->
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="root">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="__EQ__schema" type="xs:string" />
        <xs:element name="__EQ__id" type="xs:string" />
        <xs:element name="type" type="xs:string" />
        <xs:element name="default" />
        <xs:element name="title" type="xs:string" />
        <xs:element maxOccurs="unbounded" name="required" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="properties">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="id">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="type" type="xs:string" />
              <xs:element name="default">
                <xs:complexType>
                  <xs:attribute name="string" type="xs:boolean"
use="required" />
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

        <xs:element name="title" type="xs:string" />
        <xs:element name="examples">
            <xs:complexType>
                <xs:simpleContent>
                    <xs:extension base="xs:string">
                        <xs:attribute name="array" type="xs:boolean"
use="required" />
                    </xs:extension>
                </xs:simpleContent>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="id_client">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="type" type="xs:string" />
            <xs:element name="default">
                <xs:complexType>
                    <xs:attribute name="string" type="xs:boolean"
use="required" />
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="title" type="xs:string" />
    <xs:element name="examples">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:string">
                    <xs:attribute name="array" type="xs:boolean"
use="required" />
                </xs:extension>
            </xs:simpleContent>
        </xs:complexType>
    </xs:element>
    <xs:element name="receiver">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="type" type="xs:string" />
                <xs:element name="default">
                    <xs:complexType>
                        <xs:attribute name="string" type="xs:boolean"
use="required" />
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="title" type="xs:string" />
    <xs:element name="examples">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:string">

```

```

                                <xs:attribute name="array" type="xs:boolean"
use="required" />
                                </xs:extension>
                                </xs:simpleContent>
                                </xs:complexType>
                                </xs:element>
                                </xs:sequence>
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="address">
                                <xs:complexType>
                                <xs:sequence>
                                <xs:element name="type" type="xs:string" />
                                <xs:element name="default">
                                <xs:complexType>
                                <xs:attribute name="string" type="xs:boolean"
use="required" />
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="title" type="xs:string" />
                                <xs:element name="examples">
                                <xs:complexType>
                                <xs:simpleContent>
                                <xs:extension base="xs:string">
                                <xs:attribute name="array" type="xs:boolean"
use="required" />
                                </xs:extension>
                                </xs:simpleContent>
                                </xs:complexType>
                                </xs:element>
                                </xs:sequence>
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="phone">
                                <xs:complexType>
                                <xs:sequence>
                                <xs:element name="type" type="xs:string" />
                                <xs:element name="default">
                                <xs:complexType>
                                <xs:attribute name="string" type="xs:boolean"
use="required" />
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="title" type="xs:string" />
                                <xs:element name="examples">
                                <xs:complexType>
                                <xs:simpleContent>
                                <xs:extension base="xs:long">
                                <xs:attribute name="array" type="xs:boolean"
use="required" />
                                </xs:extension>
                                </xs:simpleContent>
                                </xs:complexType>

```

```

        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="type" type="xs:string" />
            <xs:element name="default">
                <xs:complexType>
                    <xs:attribute name="string" type="xs:boolean"
use="required" />
                </xs:complexType>
            </xs:element>
            <xs:element name="title" type="xs:string" />
            <xs:element name="examples">
                <xs:complexType>
                    <xs:simpleContent>
                        <xs:extension base="xs:string">
                            <xs:attribute name="array" type="xs:boolean"
use="required" />
                        </xs:extension>
                    </xs:simpleContent>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="delivery_date">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="type" type="xs:string" />
            <xs:element name="default">
                <xs:complexType>
                    <xs:attribute name="string" type="xs:boolean"
use="required" />
                </xs:complexType>
            </xs:element>
            <xs:element name="title" type="xs:string" />
            <xs:element name="examples">
                <xs:complexType>
                    <xs:simpleContent>
                        <xs:extension base="xs:string">
                            <xs:attribute name="array" type="xs:boolean"
use="required" />
                        </xs:extension>
                    </xs:simpleContent>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="goodsInOrder">

```

```

        <xs:complexType>
          <xs:sequence>
            <xs:element name="type" type="xs:string" />
            <xs:element name="default">
              <xs:complexType>
                <xs:attribute name="empty-array" type="xs:boolean"
use="required" />
              </xs:complexType>
            </xs:element>
            <xs:element name="title" type="xs:string" />
            <xs:element name="items">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="type" type="xs:string" />
                  <xs:element name="title" type="xs:string" />
                  <xs:element maxOccurs="unbounded"
name="required" type="xs:string" />
                  <xs:element name="properties">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="id_good">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="type"
type="xs:string" />
                              <xs:element name="title"
type="xs:string" />
                              <xs:element maxOccurs="unbounded"
name="examples" type="xs:string" />
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="name">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="type"
type="xs:string" />
                              <xs:element name="title"
type="xs:string" />
                              <xs:element maxOccurs="unbounded"
name="examples" type="xs:string" />
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="q_good">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="type"
type="xs:string" />
                              <xs:element name="title"
type="xs:string" />
                              <xs:element maxOccurs="unbounded"
name="examples">

```

```

                                <xs:complexType>
                                    <xs:simpleContent>
                                        <xs:extension
base="xs:unsignedByte">
                                <xs:attribute name="number"
type="xs:boolean" use="required" />
                                    </xs:extension>
                                </xs:simpleContent>
                            </xs:complexType>
                        </xs:element>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element maxOccurs="unbounded"
name="examples">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="id_good"
type="xs:string" />
            <xs:element name="name" type="xs:string"
/>
            <xs:element name="q_good">
                <xs:complexType>
                    <xs:simpleContent>
                        <xs:extension
base="xs:unsignedByte">
                            <xs:attribute name="number"
type="xs:boolean" use="required" />
                                </xs:extension>
                            </xs:simpleContent>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="examples">
    <xs:complexType>
        <xs:sequence>
            <xs:element maxOccurs="unbounded"
name="examples">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="id_good"
type="xs:string" />
                        <xs:element name="name" type="xs:string"
/>
                        <xs:element name="q_good">

```



```

        <xs:complexType>
          <xs:simpleContent>
            <xs:extension
base="xs:unsignedByte">
              <xs:attribute name="number"
type="xs:boolean" use="required" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="array" type="xs:boolean"
use="required" />
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Пример сообщения, передаваемого из OMS в Систему маршрутизации [JSON]
https://drive.google.com/file/d/1VizRbX51328ss9ZPQwmSp36eISv_yg6D/view?usp=drive_link

```
{
  "id": "order000001",
  "id_client": "client010",
  "receiver": "Устинов Н.А.",
  "address": "Москва, Кутузовский пр., 32А",
  "phone": "+79162820282",
  "status": "Подтверждён",
  "delivery_date": "31.07.2023",
  "goodsInOrder": [
    {
      "id_good": "goods0002",
      "name": "Мыло душистое",
      "q_good": 1
    },
    {
      "id_good": "goods0004",
      "name": "Полотенце пушистое",
      "q_good": 2
    }
  ]
}
```

}

Описание JSON-схемы для сообщения, передаваемого из OMS в Систему маршрутизации

https://drive.google.com/file/d/1F0lwo79_AspJMSZNFuqFcvRrYolEnlt/view?usp=sharing

```
{
  "$schema": "https://json-schema.org/draft/2019-09/schema",
  "$id": "http://example.com/example.json",
  "type": "object",
  "default": {},
  "title": "Root Schema",
  "required": [
    "id",
    "id_client",
    "receiver",
    "address",
    "phone",
    "status",
    "delivery_date",
    "goodsInOrder"
  ],
  "properties": {
    "id": {
      "type": "string",
      "default": "",
      "title": "The id Schema",
      "examples": [
        "order000001"
      ]
    },
    "id_client": {
      "type": "string",
      "default": "",
      "title": "The id_client Schema",
      "examples": [
        "client010"
      ]
    },
    "receiver": {
      "type": "string",
      "default": "",
      "title": "The receiver Schema",
      "examples": [
        "УСТИНОВ Н.А."
      ]
    },
    "address": {
      "type": "string",
      "default": "",
      "title": "The address Schema",

```

```

    "examples": [
      "Москва, Кутузовский пр., 32А"
    ]
  },
  "phone": {
    "type": "string",
    "default": "",
    "title": "The phone Schema",
    "examples": [
      "+79162820282"
    ]
  },
  "status": {
    "type": "string",
    "default": "",
    "title": "The status Schema",
    "examples": [
      "Подтверждён"
    ]
  },
  "delivery_date": {
    "type": "string",
    "default": "",
    "title": "The delivery_date Schema",
    "examples": [
      "31.07.2023"
    ]
  },
  "goodsInOrder": {
    "type": "array",
    "default": [],
    "title": "The goodsInOrder Schema",
    "items": {
      "type": "object",
      "title": "A Schema",
      "required": [
        "id_good",
        "name",
        "q_good"
      ],
      "properties": {
        "id_good": {
          "type": "string",
          "title": "The id_good Schema",
          "examples": [
            "goods0002",
            "goods0004"
          ]
        }
      }
    },
    "name": {
      "type": "string",
      "title": "The name Schema",
      "examples": [

```

```

        "Мыло душистое",
        "Полотенце пушистое"
    ]
},
"q_good": {
    "type": "integer",
    "title": "The q_good Schema",
    "examples": [
        1,
        2
    ]
}
},
"examples": [
    {
        "id_good": "goods0002",
        "name": "Мыло душистое",
        "q_good": 1
    },
    {
        "id_good": "goods0004",
        "name": "Полотенце пушистое",
        "q_good": 2
    }
]
},
"examples": [
    [
        {
            "id_good": "goods0002",
            "name": "Мыло душистое",
            "q_good": 1
        },
        {
            "id_good": "goods0004",
            "name": "Полотенце пушистое",
            "q_good": 2
        }
    ]
]
},
"examples": [
    {
        "id": "order000001",
        "id_client": "client010",
        "receiver": "Устинов Н.А.",
        "address": "Москва, Кутузовский пр., 32А",
        "phone": "+79162820282",
        "status": "Подтверждён",
        "delivery_date": "31.07.2023",
        "goodsInOrder": [
            {

```

```
{
  "id_good": "goods0002",
  "name": "Мыло душистое",
  "q_good": 1
},
{
  "id_good": "goods0004",
  "name": "Полотенце пушистое",
  "q_good": 2
}
]
}
```

Создание товара

Краткое описание метода

Метод предназначен для создания нового товара

Метод запроса: POST

Строка запроса: goods/

Блок Authorization:

Параметр	Описание	Тип данных	Обязательность
api key	ключ API	строка	+

Входные параметры:

Параметр	Описание	Тип данных	Обязательность
good_name	Название товара	Строка	+
description	Описание	Строка	нет

price	Цена	Число	нет
discount	Скидка	Число	нет
height	Габариты, высота	Число	нет
width	Габариты, ширина	Число	нет
depth	Габариты, глубина	Число	нет
photo_link	Ссылка на фото	Строка	нет

Пример запроса:

```
{
  "good_name": "Сейф",
  "description": "Классический сейф, который вы много раз видели в отелях по всему миру",
  "price": 1000,
  "discount": 20,
  "height": 300,
  "width": 500,
  "depth": 400,
  "photo_link": "https://our.universal.internet-shop.ru/photo/12345.jpg"
}
```

Выходные параметры:

В ответ сервис должен вернуть application/json ответ, содержащий параметры:

Параметр	Описание	Тип данных
good_id	идентификатор товара	Строка

Пример ответа:

```
{  
  "good_id": "12345789"  
}
```