

Подключение к Grocery – Delivery Club

Ссылки на документацию

Вся документация доступна по ссылке: <https://cloud.mail.ru/public/21wu/46UaQct9o>

1.1 `swagger.yml` – получение каталога, остатков, создание заказа и апдейт статусов через поллинг (мы ходим в партнера с заданной периодичностью)

1.2 `swagger-events.yml` – получение каталога, остатков, изменение информации по заказу через событийную модель. Можно реализовать, если необходимо обеспечить для пользователя самый актуальный состав каталога и остатков (партнер отправляет в Delivery Club, изменения по каталогу и остаткам)

2. `addon-delivery-slots.yml` – получение слотов доставки от партнеров (реализовано под СберМаркет и Утконос, можно адаптировать под любого партнера, предлагающего слоты доставки)

3. `soft-checkout.yml` – для реализации механики с софтвером (реализовано под Дикси, можно адаптировать под любого партнера, если необходимо пробрасывать заказ на кассовый аппарат)

Документация выше покрывает любой тип интеграции:

- Когда заказ собирает и доставляет сам партнер. Пример: Самокат, Быстроном;
- Когда заказ собирает партнер, а доставляет курьер Delivery Club. Например: Вкусвилл, Верный;
- Когда заказ принимает сборщик Delivery Club и доставляет курьер Delivery Club. Например: Магнит, Дикси, Мираторг;
- Когда у партнера есть слоты доставки и он самостоятельно собирает и доставляет заказ: Сбермаркет, Утконос.

1 – Базовая имплементация API

Включает в себя реализацию 4 групп методов:

1. **Authentication** – включает метод для создания токена авторизации. Вызывается прежде, чем будет вызван любой другой запрос;
2. **Order** – набор из трех обязательных методов: на создание заказа, опроса статусов по сделанным заказам и изменение статуса заказа;
3. **Catalog** – включает методы на получение каталога товаров: категорий и продуктов;
4. **Stock** – содержит метод для получения информации по остаткам товаров в каждом магазине;

Исходный Swagger с детальным описанием методов можно скачать по ссылке: <https://cloud.mail.ru/public/4XgB/2ohXsSiCE>. Так же можно посмотреть Swagger здесь: <https://app.swaggerhub.com/apis-docs/dsmbdgv/integration-api/1.2.0/#/>

Подробнее о методах:

1.1 – Методы группы Order

1. `POST/stores/{storeid}/orders` – используется, чтобы создать заказ в определенном магазине;
2. `PUT/stores/{storeid}/orders/{orderid}/status` – вызывается, когда заказ отменяется на стороне Delivery Club (*status = canceled*) и когда заказ успешно доставлен (*status = delivered*).
3. `GET/stores/{storeid}/orders/{orderid}` – используется после создания заказа для получения:
 - Статусов по заказу;
 - Измененного состава и суммы, в случае корректировки.

Примечание 1. В случае если вы измените состав заказа (чего то не хватило, что то испортилось), то в методе мы должны получить от вас признак *"changedOnFormed": true*, в массиве с *"positions"* измененный набор товаров и в *"total"* измененную цену.

Примечание 2. Если произошла корректировка весового товара (содержащего квант) – нужно передавать дробный *"quantity"*. Пример: клиент купил 4 кванта курицы по 500 грамм. Итого: 2 кг курицы, набранной 4-мя квантами. Пикер собрал 1.9 кг – это 3.8 кванта – его и нужно передать, вместе с финальной суммой, а DC вернет деньги за 0.2 кванта, которые не приедут.

После получения этих данных DC автоматически отправит пользователю пуш уведомление и вернет деньги за недоставленные товары или доспешет деньги, если корректировка в большую сторону;

1.2 – Методы группы Catalog

1. `GET/stores/{storeid}/catalog` – для получение категорий и продуктов одним запросом;

2. **GET/stores/{storeid}/catalog/categories** и **GET/stores/{storeid}/catalog/products** – используется если есть желание разделить получение категорий и продуктов на два отдельных запроса:

Примечание 1: исторически, Delivery Club не умеет работать с копейками. Поэтому на стороне партнера, перед передачей цен по продуктам в Delivery Club требуется производить округление. Например, до ближайшего целого или в большую/меньшую сторону;

Примечание 2: по умолчанию забор каталога осуществляется 1 раз в час. При желании со стороны партнера, можем забирать каталог с меньшей частотой;

Примечание 3: на стадии MVP для получения каталога используется “поллинг” подход, когда Delivery Club с заданной частотой опрашивает метод(ы) каталога. В дальнейшем партнеру предлагается перейти на “событийную модель”, когда партнер сам инициирует обновления каталога, а Delivery Club получив сигнал об изменениях приходит за обновлением. Swagger с описанием “событийной модели” здесь: <https://cloud.mail.ru/public/3SqN/2u9BH9fPw>

1.3 – Метод группы Stock

1. **GET/stores/{storeid}/stock** – используется для получения информации об остатках товаров по магазину.

Примечание 1. частоту забора остатков, равно как и каталога, можно гибко настраивать. Максимально возможная частота 1 раз в минуту.

Примечание 2: на стадии MVP для получения остатков используется “поллинг” подход, когда Delivery Club с заданной частотой опрашивает метод остатков. В дальнейшем партнеру предлагается перейти на “событийную модель”, когда партнер сам инициирует обновления остатков, а Delivery Club получив сигнал об изменениях приходит за обновлением. Swagger с описанием “событийной модели” здесь: <https://cloud.mail.ru/public/3SqN/2u9BH9fPw>

2 – Контентная подготовка

Этот шаг включает в себя:

1. Составление плана запуска вместе с партнером;
2. Отрисовка полигонов доставки (либо автоматическая отрисовка на стороне Delivery Club, если у партнера нет GEO полигонов);
3. Создание маппинг категорий для мобильного и веб приложения (делает Delivery Club);
4. Заведение магазинов внутри админок (делает Delivery Club);

Примечание 1: для упрощения запуска магазинов, партнеру необходимо подготовить таблицу со столбцами:

- store_id (идентификатор магазина в котором будем создавать заказ)
- city
- address
- phone
- lat/long
- schedule (график работы)
- geo-json (если есть)

Примечание 2: ознакомьтесь дополнительно с простыми правилами оформления контента, что бы в приложении DC продукты выглядели аккуратно: https://docs.google.com/presentation/d/18YxD9FPeAb4s7h_wj8JKAuaz9fVkrj8LMeCc9hdKM0Q/edit?usp=sharing

3 – Тестирование

1. Определяем вместе с партнером магазин на котором можно производить тестирование;
2. Проверяем всевозможные кейсы взаимодействия: обновление каталога, обновление остатков, создание заказа, получение статусов по заказу, отмены заказов с двух сторон, корректировки весовых и не весовых товаров, получение клиентом уведомлений, выдача заказа курьеру, доставка заказа до клиента и т.д

Часто задаваемые вопросы

	Вопрос	Ответ
1	Какова механика работы с весовым товаром	В случае если вы измените состав заказа (чего то не хватило, что то испортилось), то в методе GET/stores/{storeid}/orders/{orderid} мы должны получить от вас признак "changedOnFormed": true, в массиве с "positions" изменённый набор товаров и в "total" изменённую цену. Если произошла корректировка весового товара (содержащего квант) – нужно

		<p>передавать дробный "quantity". Пример: клиент купил 4 кванта курицы по 500 грамм. Итого: 2 кг курицы, набранной 4-мя квантами. Пикер собрал 1.9 кг – это 3.8 кванта – его и нужно передать, вместе с финальной суммой, а DC вернет деньги за 0.2 кванта, которые не приедут.</p> <p>После получения этих данных DC автоматически отправит пользователю пуш уведомление и вернет деньги за недоставленные товары или доспешет деньги, если корректировка в большую сторону;</p>
2	Если товара нет в наличии как мы это указываем в заказе? Есть ли возможность удалить позицию? Или надо звонить в колл-центр?	Товары удалять можно, главное – вернуть измененный состав в api метод GET/stores/{storeId}/orders/{orderId}. Мы отправим пользователю пуш и вернем часть денег за отсутствующие товары. В КЦ звонить не нужно.
3	Если товара нет в наличии, кто оповещает клиента, возможна ли замена по согласованию с клиентом?	Да, замены производить можно, но так, что бы обновленная сумма заказа не превысила изначальную стоимость заказа на 10%. Например, изначальный заказ был на 1000 рублей, позволено добавлять/заменять товары в пределах 1100 рублей. В методе на обновление состава GET/stores/{storeId}/orders/{orderId} заказа вернуть обновленный состав.
4	Клиент не принял товар/часть товара/ не вышел на связь/возврат товара - что будет с товаром физически (куда его денет курьер), будет ли оплата?	Если клиент не принял заказ в течение 15 минут, то курьер вернет заказ обратно в магазин, клиенту вернут деньги по запросу. Если клиент не принял часть товара (например, что то было испорчено), то клиенту вернут деньги за часть товара и выдадут sogu-промокод за счет DC. Испорченный товар остаётся у клиента, курьер его не возвращает в магазин. Партнеру будет выставлен штраф за не качественный товар.
5	Продажа заморозки/охлажденки - как мы должны упаковать, рекомендации?	Желательно положить такие товары в отдельный целлофановый пакет. Курьер разделит горячие и холодные блюда в своей термо сумке, что бы холодное доехало не растаявшим, а горячее не остыло.
6	Габариты/размеры пакета (чтобы поместился в сумку)	Точно известно, что 3 пятилитровых бутылки в сумку не поместятся. Соответственно объем заказ должен быть меньше 15 литров.
7	Большой заказ - как определяют, что необходим второй курьер? (риски: задержка доставки заказа, порча охлажденки и заморозки)	На текущий момент если курьер понимает, что не увезет заказ один (слишком тяжелый/объемный), то он запрашивает подкрепление в диспетчерской и ему приходит на помощь второй курьер. В дальнейшем на уровне системы мы реализуем ограничение на вес корзины
8	Ограничение по весу заказа?	Что бы мог унести 1 курьер. ~10 - 15 кг максимум
9	Разделение товара по товарному соседству в разные пакеты? (в заказе горячий суп и мороженое)	Да, такие товары разделять нужно в разные пакеты. В курьерской термосумке, они будут лежать в разных отсеках
10	Герметизация заказа с нашей стороны нужна? (минимизация пропачи части товара)	Не обязательно. На практике герметизацию сейчас используют только Аптеки. Для grosery товаров это избыточно.
11	Механика работы со скидками/акциями? (в ручном режиме или интеграция, на категорию, на каждый товар?)	В методе каталога GET/stores/{storeId}/catalog, есть возможность указать оригинальную стоимость товара и дисконтированную. При наличии дисконтированной цены на товар, мы выведем его в специально скидочном блоке на видном месте сверху каталога. Так же можем собрать отдельную категорию со скидками и поместить её в каталоге продукта.
12	Фото - разрешение минимальное/ фон, соотношение сторон и т.д.	Передавайте самое качественное изображение которое есть. Идеально соотношение сторон 1 к 1. Но если такой возможности нет, то подойдет соотношение сторон как у прямоугольника. Например, у Вкусвилла изображения товаров прямоугольные. А у Самоката квадратные (1 к 1) и то и другое выглядит нормально.

13	Фото - где должны лежать, как обозначить, что от определенного товара (код/наименование?) (ссылка с облака или на сервере?)	Необходимо передать imageUrl в объекте с данными о продукте.
14	Для каждой категории необходимо сделать фото или можете свое выставить?	Для каждой категории первого уровня мы самостоятельно подготовим изображения.
15	Заказы будут приходить также как и приходят на еду? (письмо на почту, оповещение на личный кабинет)	Заказы будут приходить через АПИ. Далее заказ нужно будет отображать в том софте который у вас есть. Если софта для приема заказов нет, то мы можем выдать вам для этого специальное приложение Vendor App/Picker App в котором заказ можно будет принимать.
16	Доступность ассортимента - я правильно понимаю что вы будете подтягивать остатки каждого магазина и если остаток будет равен "0" - то позиция отображаться в кабинете для клиента не будет?	Верно. Если товар закончился на стоках, то клиенту в каталоге он показываться не будет.
17	Наименование категорий и подкатегорий задаем мы и соответственно передаем через API? Например: мы планируем заводить 25 категорий, в каждой из них в среднем по 10 подкатегорий и уже в подкатегории сам товар.	Да, наименование категорий и подкатегорий задаёте вы и передаёте по API
18	Возможно ли помимо интеграции, добавлять позиции вручную в кабинете приложения Vendor App? Объясню: мы планируем начать с 20 магазинов, но только в 5-ти магазинах есть кулинария, и чтобы не интегрировать остатки каждой точки, мы хотели бы загрузить одну стандартную матрицу на все точки, а где есть кулинария (это всего 5 магазинов)добавлять вручную?	На текущем этапе нет такой возможности нет. Весь каталог должен передавать только по API. В описанном кейсе нужно поступить так: передавайте нам абсолютно весь каталог, а для тех магазинов где не будет кулинарии, она просто не будет отображаться, потому что залогом того, что товар отображается в каталоге, является наличие остатков по товару.
19	Недовоз Товара - как решается вопрос с клиентом и с нами? (например наш сотрудник недоложил (человеческий фактор)или курьер вытащил)	Клиенту, по обращению в наш КЦ, возвращают часть денег за недоставленный товар. Выдают согу промокод за счет DC
20	В презентации указано, что наименование должно состоять "Идеально 26 знаков", "Учитывая, что наименование больше 56 знаков уместится только на отдельном экране", я правильно понимаю, что при быстром просмотре наименование до 56 знаков уместится и будет отражено, если больше, то только в карточке товара при ее раскрытии, а 56 знаков, то есть неполное название, отобразится на нераскрытой карточке товара? Но я нашел и позицию где в наименовании 92 знака и название отображается все целиком на общем экране вместе с фото У Быстронома. КАКОЕ МАКСИМАЛЬНОЕ КОЛИЧЕСТВО ЗНАКОВ ДЛЯ ОТОБРАЖЕНИЯ ВСЕГО НАЗВАНИЯ НА ОБЩЕМ ЭКРАНЕ?	52 знака идеально – в этом случае всё название будет в три строки. На карточке товара в боттом щите – поместится до 60 знаков. В случае если название будет длиннее оно обрежется и остаток названия будем помечен многоточием.
21	Вес и объем мы отражаем в отдельной ячейке?	Вес и объем лучше отображать в названии товара (там где релевантно). Также вес необходимо отображать в рамках мета-полей описания товара. Дополню: вес товара (литры, килограммы, граммы) – нужно склеивать с названием товара. Что бы на карточке товара было видно, что например "Молоко деревенское, 1л" или "Курица охлажденная, 600 г".
22	Единица измерения, Штука или кг или квант тоже в отдельной ячейке?	Кванты передаются в рамках поля "grammar", если такого поля нет для товара, то мы товар считаем штучным
23	Просьба прописать конкретно те поля которые мы должны передать, чтобы не было вопросов к Вам со стороны закона	Необходим следующий набор полей: 1. id – Идентификатор продукта. Всегда строка, не число (но в строке может быть записано число). 2. name – Наименование продукта 3. imageUrl – URL на изображение 4. categories – Массив идентификаторов категорий 5. vat – НДС 6. price – Цена продукта без скидки 7. discountPrice – Цена продукта со скидкой 8. grammar – Квантовка товара 9. weight – Вес в граммах 10. meta – Мета товара в формате Ключ - Значение 11. brand – Торговая марка
24	Какие символы недопустимы при заполнении Составы/метаданных	Прикладываю гайд по оформлению контента: https://docs.google.com

		<p>com/presentation/d/18YxD9FPeAb4s7h_wj8JKAuaz9fVkrj8LMeCc9hdKM0Q/edit?usp=sharing. Пожалуй всё что имеет тип string допустимо в названии, а html теги и другие артефакты верстки нет. Единственное, ознакомьтесь с инструкцией по контенту там много полезного.</p>
25	<p>В каком формате передается информация по кванту/шагу заказа? планируем передавать только для весового товара шаг 1КГ указываем так или 1000 грамм?/ 0,5 кг или 500 грамм?</p>	<p>Кванты передавать в граммах. В данном случае 1000 г</p>