Описание API

Перечень методов API ProCart

- **/ping** - служебный метод для проверки связи, состояния лицензии, валидности токена.
- **/getmenu** - метод для получения меню (классификация, блюда и их свойства, остатки, схемы модификаторов и комбо).
- **/validate** - метод для проверки возможности создания заказа (заказ НЕ создается).
- **/postorder** - метод для создания заказа в кассовой системе.
- **/getorders** - метод для получения информации о заказах.
- **/close** - метод для закрытия заказа.
- **/internal/print_info** - служебный метод для получения длинных комментариев (используется в макетах печатных форм r_keeper).

Документация к API - ссылка Токен API: 28bb0d65-d1a3-447d-a3eb-e4961b0ffd35

Все запросы отправляются в «ip:port/api/v3» Например 127.0.0.1:11111/api/v3/ping

Важно! Все методы возвращают ответы в кодировке UTF-8

Особенности использования методов API

ping

Метод **/api/v3/ping** не является обязательным и не приводит ни к каким действиям на стороне кассовой системы, но в то же время он очень полезен. С его помощью можно:

- Проверить доступность ProCart
- Проверить связь между модулем ProCart и кассовым сервером
- Получить информацию о версиях ProCart и кассовой системы r_keeper
- Получить информацию о наличии лицензии на модуль ProCart
- Проверить валидность токена используемого для работы

Рекомендуется выполнять запрос **/ping** при начале работы.

---

getmenu

Метод **/*api/v3/getmenu** позволяет получить информацию о блюдах и их остатках, доступных для продажи во внешней системе. При большом количестве блюд, ответ на данный запрос может занимать значительное время (500-2000 мс), поэтому рекомендуется производить кеширование ответов на стороне внешней системы. Модуль ProCart так же кеширует меню на своей стороне и проверяет обновления с периодичностью 10-60 секунд. Поэтому для внешней системы вызывать метод **/getmenu** чаще чем 1 раз в минуту не имеет особого смысла.

validate

Метод **/api/v3/validate** так же не является обязательным, но его использование крайне рекомендовано. Данный метод позволяет провалидировать содержимое корзины. Т.е. убедиться что такой заказ можно создать в кассовой системе в текущий момент времени. Данный метод так же нужен для корректного отображения «корзины» во внешней системе. Т.к. в зависимости от настроек r_keeper и дополнительных параметров заказа («категория заказа», стол, скидка и т.п.) кассовая система может изменять стоимость блюд в заказе. **Важно!** Вызов данного метода не приводит ни к каким событиям на кассе (заказ не создается).

---

postorder

Метод **/api/v3/postorder** создает заказ на кассе r_keeper. Если заказ создан успешно, то в ответе на запрос вернется GUID созданного в системе r_keeper заказа, используя который в дальнейшем можно отслеживать изменения состояния этого заказа используя метод **/getorders Важно!** Если логика внешней системы подразумевает оплату заказа, то перед тем как вызвать метод **/postorder** необходимо провести оплату заказа (а сумму к оплате взять из ответа на запрос **/validate**). И только после успешной оплаты создавать заказ методом **/postorder**, не забыв при этом передать данные об оплате.

---

getorders

Оба метода **/api/v3/getorders** получают информацию о заказах.

- В методе **POST** нужно передать токен, а так же GUID или список GUID'ов заказов. В ответ придёт информация по всем открытым заказам.
- В методе **GET** нужно передать только токен. В ответ придёт информация информация по всем открытым через ProCart заказам.

---

close

Метод **/api/v3/close** является необязательным методом. Может потребоваться для закрытия заказа, если в ProCart.yaml в группе «r_keeper_references» указана настройка - close_paid_order: false.

Типовая схема запросов

В итоге, для того что бы создать заказ на кассе, внешняя система должны последовательно выполнить запросы:

1. Убедиться в доступности шлюза ProCart (**/ping**).
2. Получить список блюд доступных для продажи (**/getmenu**).
3. Убедиться в возможности создания заказа с нужным перечнем блюд в текущий момент времени (**/validate**).
4. Создать заказ (**/postorder**).
5. При необходимости отслеживать дальнейшие изменения статуса заказа (**/getorders**).

Важно! При создании заказа, через параметр **callback_url** возможно указать адрес, куда будет отправляться состояние заказа при изменении его статуса.

Заказ может иметь статус:

- «0»=open - заказ создан в кассовой системе и с ним ведётся работа
- «1»=close - закрыт выполнен рассчёт по заказу и выбит чек
- «2»=bill - в заказе выбит пречек или чек намерения
- «3»=deleted - заказ удалён в кассовой системе
- «4»=deferred - заказа создан в кассовой системе как черновик
- «5»=new - новый заказ

From:
https://wiki.carbis.ru/ - **База знаний ГК Карбис**

Permanent link:
https://wiki.carbis.ru/external/procart/02_procart_api/01_api?rev=1762922763

Last update: **2025/11/12 04:46**