Интеграции с программами лояльности

- Схема интеграции
- SailPlay
  - Настройка
  - Сценарий идентификации покупателя
  - Запрос промокода
  - Процесс ‘Расчёт скидок’
  - Процесс ‘Возврат товара’
  - Список запросов в SailPlay
- Kilbil
  - Настройка
  - Сценарий идентификации покупателя
  - Запрос промокода
  - Процесс Расчёт скидок
  - Процесс Возврат товара
  - Список запросов в KilBil
  - Выгрузка справочника товаров в KilBil

Схема интеграции

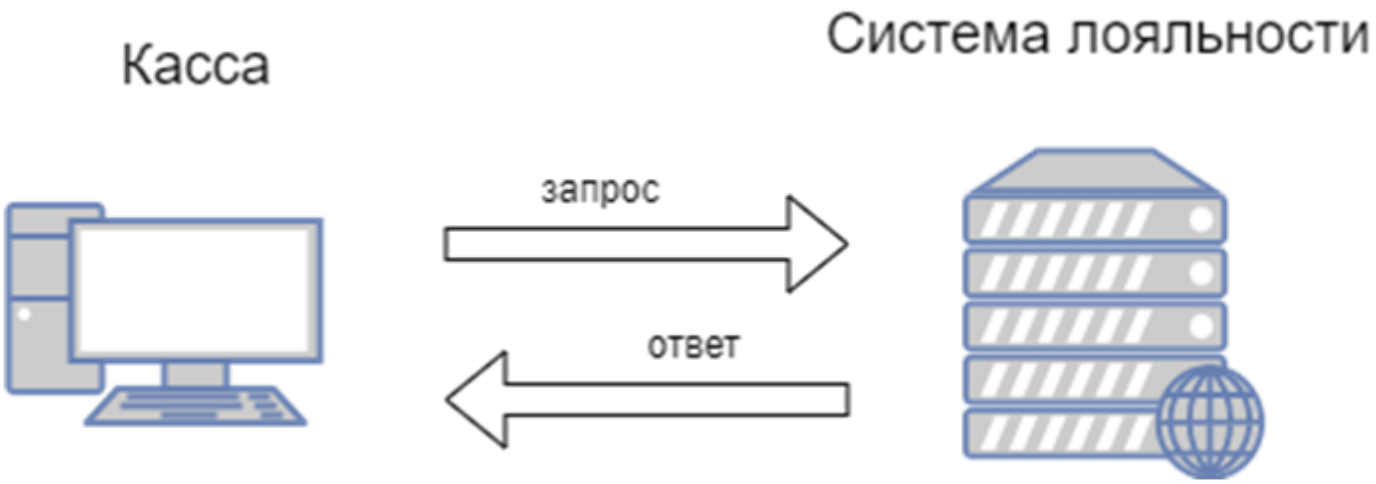
В последние годы всё чаще для расчёта скидок применяются системы лояльности. Система лояльности – это внешняя программа, которая предоставляет различные средства для привлечения и удержания клиентов с помощью поощрений в виде скидок или бонусов. При использовании системы лояльности скидки рассчитывает не касса, а система лояльности.

Мы проанализировали несколько систем лояльности, также учли собственный опыт разработки подобной системы. И выяснили, что обмен рассмотренных систем с кассой выполняется по очень похожим схемам. Запросы называются по-разному, параметры разные, но суть одинаковая. А раз так, то появилась идея реализовать в кассе методы обмена с абстрактной системой, а интеграцию с конкретной системой делать в рамках абстрактных методов. Тогда для интеграции достаточно будет реализовать только запросы конкретной системы лояльности. Такое решение позволит быстрее делать интеграции и повысит устойчивость самой кассы.

Принципы интеграции кассы с системой лояльности.

1. Обмен включается на уровне магазина. В карточке магазина указывается, что магазин является участником программы лояльности с некоторой системой. Это условие нужно для подключения магазинов к программе лояльности по мере готовности. Также, если в компании применяются несколько систем лояльности, то данный принцип позволяет выбрать систему лояльности для конкретного магазина.
2. Скидки рассчитываются только системой лояльности. Если магазин подключён к системе лояльности, то отключаются все скидки и акции, которые в ином случае были бы применены кассой самостоятельно. Этот принцип введён из-за непонятной схемы взаимодействия скидок в разных системах. Невозможно заранее определить механизм и последствия воздействия скидок в кассе на скидки в системе лояльности, и наоборот. И как следствие, объяснение покупателю, почему получилась именно такая цена, становится слишком сложным.
3. Отсутствие или временное прерывание связи с системой лояльности не прерывает работу кассы. Касса работает в любом случае, только скидки не рассчитываются.

Обмен кассы с системой лояльности выполняется по схеме Запрос-Ответ. Касса формирует и передаёт запрос. Система лояльности обрабатывает запрос и возвращает ответ. Инициатором всегда выступает касса.



Реализовано три группы запросов-ответов.

	Ответ
Идентификация покупателя	
Проверить регистрацию покупателя с таким идентификатором (номер телефона или номер карты)	- да, такой покупатель зарегистрир - нет такого покупателя
Зарегистрировать нового покупателя	- выполнено
Расчёт чека	
Рассчитать скидки по этому чеку	- чек с новыми ценами
Рассчитать число баллов, которое покупатель может потратить на оплату чека	- число баллов
Сохранить чек в базе покупок	-выполнено
Возврат	
Найти чек продажи по этим реквизитам	- чек продажи
Оформить возврат этих позиций	- выполнено

Первая группа применяется для идентификации покупателя в системе лояльности. Кассир запрашивает у покупателя информацию для его идентификации. Это может быть номер дисконтной карты или, что удобнее, номер телефона. Вводит (или сканирует) данные на экране кассы, касса отправляет запрос ‘Проверь, имеется ли покупатель с такими

данными’. Система лояльности выполняет поиск в базе покупателей и возвращает ответ. Если такого покупателя нет, то кассир может предложить покупателю зарегистрироваться. При согласии отправляет запрос на регистрацию нового покупателя.

Во вторую группу входят запросы по расчёту скидок и баллов по чеку. Касса отправляет чек и получает обратно чек с новыми ценами. Как система лояльности вычисляет новые цены, какие применяет скидки, касса не знает. Задача кассы – принять ответ и внести изменения в чек. После оплаты касса повторно передаёт чек, чтобы система лояльности зарегистрировала продажу.

Последняя группа запросов связана с возвратами. Один запрос нужен для получения чека продажи. Второй – для регистрации возврата.

SailPlay

Настройка

В карточке компании указывается токен и основной код, далее по тексту он будет называться КОД ИЗ ГЛОБАЛЬНЫХ НАСТРОЕК. Этот код используется во всех запросах к SP.

Также задаются адрес сервера SP для запросов и время (в секундах) ожидания ответа от SP на запрос кассы. Если за указанное время SP не ответит, то касса будет обрабатывать эту ситуацию как ошибку обмена (отсутствие связи).

В карточке магазина указывается признак наличия обмена с SP. Обмен касс магазина с SP включается только в том случае, если будет указан данный признак.

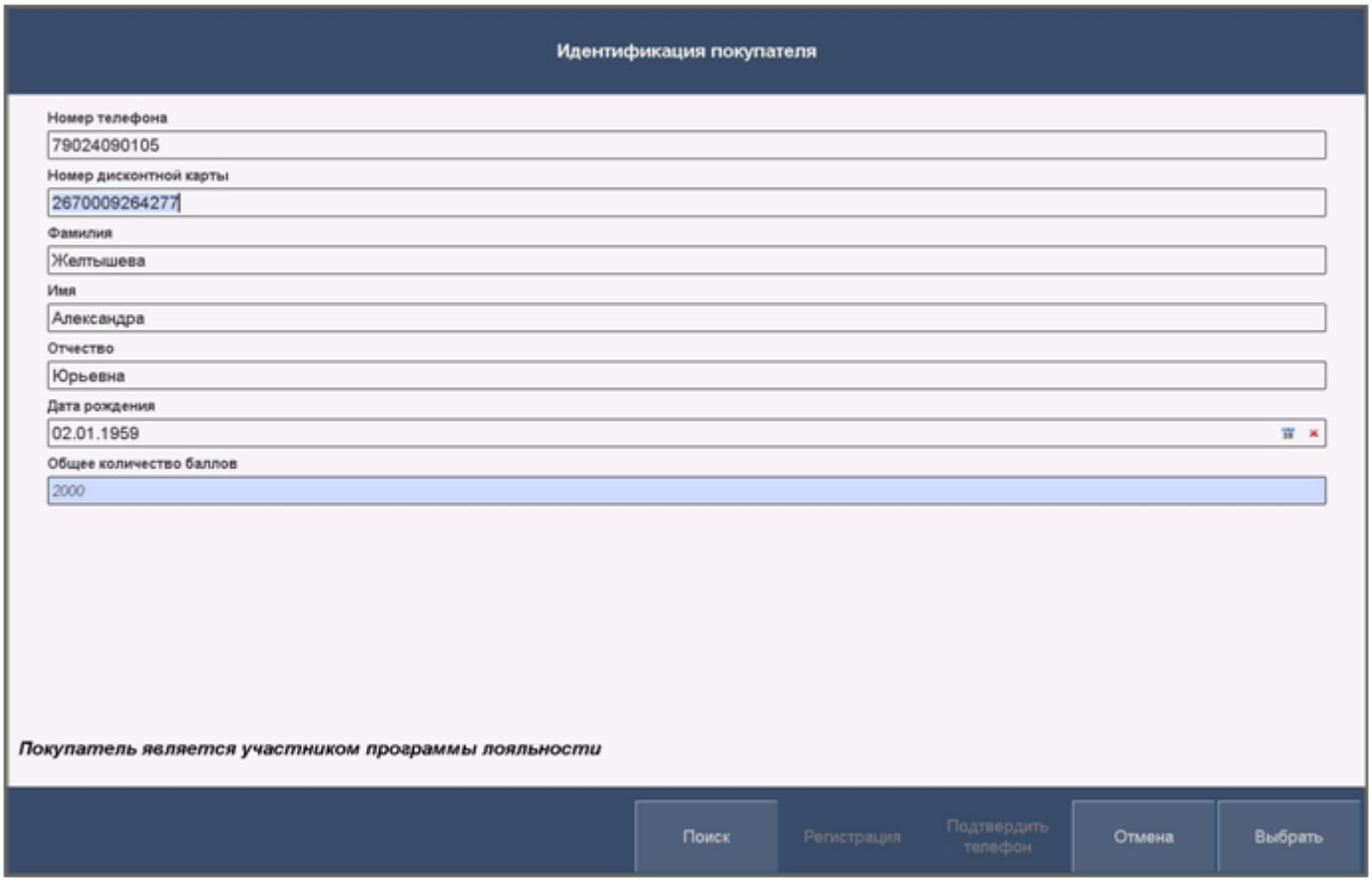
В настройке интерфейса кассира следует привязать к горячим клавишам две команды: ‘*Идентификация покупателя SP*’, ‘*Запрос промокода’*. Эти команды используются для запуска соответствующих процессов.

SailPlay

Сценарий идентификации покупателя

Процесс запускается по команде ‘*Идентификация покупателя SP*’. По окончании процесса касса получает идентификатор покупателя. Если идентификатор не будет заполнен, то считается, что покупатель не участвует в программе лояльности.

На экране отображается форма с данными покупателя, информационной строкой, кнопками для выбора действий.



Поля:

- Номер телефона (начинается с 7)
- Номер карты – можно сканировать ШК, можно ввести вручную часть номера

- Фамилия
- Имя
- Отчество
- Дата рождения
- Число баллов (для просмотра)
- Строка с сообщением для кассира

Кнопки:

- Поиск
- Регистрация
- Отмена
- Подтвердить телефон
- Выбрать

Вся дальнейшая работа внутри процесса выполняется в данной форме. Форма закрывается только при выходе из процесса. Промежуточные результаты отображаются в полях формы и в информационной строке.

Сначала кассир вводит один из идентификаторов покупателя: номер телефона или номер карты. В терминах SP это user_phone и origin_user_id. Номер телефона начинается с цифры 7.

Предполагается, что в будущем останется только один идентификатор – номер телефона. Но в переходный период идентификатором также является номер дисконтной карты в Домино. Номер карты – это 13-значное число в формате EAN13. Кассир может либо сканировать ШК карты, либо ввести короткий номер карты. Код EAN13 получается из номера карты путём добавления префикса 267, далее необходимое число нулей, введённый код, контрольный символ.

Основные сценарии работы кассира

1. Покупатель является участником программы лояльности

Если покупатель говорит, что он является участником программы лояльности, то кассир выбирает кнопку ‘*Поиск*’. Касса отправляет в SP запрос users/info. SP по номеру телефона (или номеру карты) ищет в своей БД запись участника программы. Если поиск завершается успешно, то SP в ответе на запрос возвращает информацию об участнике программы. Касса

заполняет остальные поля формы данными из ответа. Далее проверяется наличие значения в поле Телефон. Если поле не заполнено, то в информационной строке отображается тест: ‘*Требуется подтверждение телефона*’. Если телефон заполнен, то в информационной строке появляется сообщение ‘*Покупатель является участником программы лояльности*’. Если SP присылает отрицательный ответ (подходящая запись в БД не обнаружена), то в информационной строке отображается ответ из SP.

Далее кассир анализирует информацию на экране и действует по одному из следующих сценариев.

- *1.1. Участие подтверждено, телефон имеется. В строке сообщения находится текст ‘Покупатель является участником программы лояльности’*

Кассир нажимает кнопку ‘*Выбрать*’. Форма закрывается. Идентификатор покупателя определён.

- *1.2. Участие подтверждено, не указан телефон. В строке сообщения находится текст ‘Требуется подтверждение телефона’*

Кассир вводит номер телефона и нажимает кнопку ‘*Подтвердить телефон*’.

Сначала касса отправляет запрос users/info для проверки, что данный телефон ещё не зарегистрирован. Если ответ положительный, то в информационной строке отображается текст ‘*Покупатель с таким телефоном уже зарегистрирован*’. Выполняется возврат в форму. Кассир либо вводит другой номер телефона, либо отменяет идентификацию.

Если ответ отрицательный (т.е. в базе SP нет регистрации данного телефона), то касса отправляет в SP запрос send/sms-code для подтверждения номера. Если ответ отрицательный, то в информационную строку записывается сообщение об ошибке. Выполняется возврат в форму.
В случае положительного ответа приходит код подтверждения. Отображается новая форма для запроса кода. В форме одно поле ‘*Введите код*’ и две кнопки: ‘*Проверить*’, ‘*Отмена*’. Покупатель называет код, полученный по sms, кассир вводит этот код и нажимает ‘*проверить*’. Касса сверяет введённый код с кодом, полученным в ответе на запрос.
Если коды совпадают, то касса отправляет в SP запрос users/update. Если ответ положительный, то в информационной строке отображается ‘*Телефон подтверждён*’.

Выполняется возврат в основную форму. В информационной строке появляется сообщение ‘П*окупатель является участником программы лояльности*’.

Если получен отрицательный ответ на запрос, то выполняется возврат в форму, в информационной строке отображается сообщение об ошибке.

Если код, названный покупателем, не совпадают с кодом, полученным по запросу, то в форме для запроса отображается ‘*неверный код’*. Кассир может повторно ввести код и нажать ‘*проверить*’. Если кассир нажимает ‘*Отмена*’, то выполняется возврат в основную форму без подтверждения телефона. В информационной строке остаётся текст ‘ *Требуется подтверждение телефона*’.

- *1.3. Участие не подтверждено. Текст в информационной строке содержит описание ошибки*

Дальнейшие действия либо по ветке 2, либо по ветке 3.

2. Покупатель не является участником программы лояльности и не хочет участвовать

Кассир выбирает ‘*Отмена*’. Форма закрывается, идентификация завершается без заполнения идентификатора покупателя.

3. Покупатель не является участником программы лояльности и хочет зарегистрироваться

Кассир заполняет в форме остальные параметры. Обязательно должны быть заполнены телефон, Имя, Фамилия. Отчество и дата рождения не обязательны к заполнению. Номер карты для регистрации не нужен и будет очищен.

Далее выполняются действия по проверке телефона в базе SP и его подтверждения. Подробно эти действия описаны в ветке 1.2. Алгоритм действий такой же, но после успешного подтверждения телефона регистрация продолжается. Касса отправляет в SP запрос на регистрацию users/add. Если в ответе будет ошибка, то отображается текст ошибки в информационной строке. Выполняется возврат в основную форму. В случае положительного ответа форма закрывается. Идентификатор покупателя определён.

Запрос промокода

Процесс запускается по команде ‘*Запрос промокода*’. По окончании процесса касса получает название промокода. Если название промокода не заполнено, то считается, что промокода нет.

На экране отображается форма с одним полем ‘Промокод’ и двумя кнопками ‘*Подтвердить*’ и ‘*Отмена*’. Никакие проверки введённой строки не выполняются.

Промокод будет применён в запросах в SP для расчёта скидок.

Никаких ограничений на доступ к команде нет. Доступность команды не зависит от наличия обмена с SP или идентификации покупателя.

Промокод на чек может быть только один. Повторный ввод промокода удаляет предыдущий.

Процесс ‘Расчёт скидок’

Данный процесс является неотъемлемой частью основных кассовых процессов заполнения и оплаты чека.

Условно процесс делится на четыре этапа:

- Регулярный расчёт скидок по чеку
- Запрос у покупателя количества баллов для оплаты чека
- Финальный расчёт чека
- Создание чека в SP

Этапы процесса выполняются при следующих условиях:

- включён обмен с SP (заполнены параметры в карточке компании, по этим параметрам получен токен для запросов в SP, имеется связь с SP)
- магазин является участником программы лояльности (заполнен параметр в карточке магазина)
- чек не является оформлением покупки по интернет-заказу (кассир не запрашивал номер интернет-заказа)

Для запроса баллов и расчёта скидок за баллы дополнительно требуется успешная идентификация покупателя. Если покупатель не идентифицирован как участник программы лояльности, то баллы не запрашиваются.

Регулярный расчёт скидок по чеку

После каждого изменения содержания чека (добавление и удаление товаров, изменение количества) касса отправляет в SP запрос marketing-actions/calc. Число баллов в запросе равно 0, чтобы SP при расчёте скидок не учитывал баллы. В случае положительного ответа от SP касса разбирает ответ и изменяет фактические цены товаров. Если SP не ответил или вернул отрицательный ответ, то цены не меняются.

При отсутствии связи с SP на экране красным цветом загорается соответствующий сигнальный символ. Если связь прервалась после предыдущего успешного запроса, то на экране высвечивается сообщение для кассира ‘*Сервер лояльности недоступен*’. Если на предыдущий запрос также не было ответа, то повторно сообщение не высвечивается.

Запрос баллов

Выполняется только при условии успешной Идентификации покупателя.

В процессе Идентификации SP вернул число баллов. Эти баллы можно потратить для оплаты чека – на товары может быть рассчитана дополнительная скидка в рублях, равная числу баллов.

После перехода в режим оплаты на экране появляется окно с предложением ввести количество баллов для оплаты чека. По умолчанию в поле находится имеющееся значение баллов. Кассир по согласованию с покупателем может уменьшить это число. В форме имеются две кнопки: ‘*Подтвердить*’ и ‘*Отмена*’. Если кассир нажал ‘*Подтвердить*’, то число баллов для оплаты принимается равным введённому значению. Выбор кнопки ‘*Отмена*’ означает отказ от списания баллов.

Финальный расчёт чека

Выполняется только при условии успешной Идентификации покупателя, после запроса баллов. Касса отправляет в SP финальный запрос для расчёта скидок marketing-actions/calc. В запросе должен быть заполнен тег с числом балов, чтобы SP при расчёте скидок учитывал баллы. Значение числа баллов берётся из ответа на предыдущий запрос (calc) баллов в SP.

В случае положительного ответа от SP касса разбирает ответ и изменяет фактические цены товаров. Если SP не ответил или вернул отрицательный ответ, то касса ничего не делает.

Если покупатель отказывается от оплаты и возвращается в редактирование чека, то следует сбросить расчёт с учётом баллов. Для этого касса отправляет в SP запрос на расчёт чека с нулевым количеством баллов.

Подтверждение оплаты чека

Выполняется после полного оформления чека, когда покупатель оплатил покупку и чек отправлен на ФР.

Касса отправляет в SP запрос purchases/new для создания покупки. Чтобы не передавать все товары, используется идентификатор корзины в базе SP (cart_id). Этот идентификатор возвращает marketing-actions/calc.

Если SP вернул ошибку, то повторяем расчёт чека (marketing-actions/calc) и ещё раз создание покупки (purchases/new). Если опять ошибка, то чек сохраняется в списке неотправленных чеков.

Если связь с SP отсутствует, то чек также сохраняется в списке неотправленных чеков.

Список неотправленных чеков нужен только для тех чеков, по которым был выполнен расчёт скидок, покупатель оплатил чек, но информация о покупке в SP не была передана. Т.е. SP расчёты выполнил, но запрос на создание покупки не получил.

Подтверждение покупки выполняет SP в автоматическом режиме. Касса запрос на подтверждение не отправляет.

SailPlay

Процесс ‘Возврат товара’

Покупатель оформляет возврат товаров в кассе. Предполагается, что это отказ от части позиций, без добавления новых позиций.

Кассир вводит реквизиты чека продажи. Поиск чека сначала выполняется в базе кассы. Если чек не обнаружен, то выполняется запрос покупки в SP purchases/get. Если SP вернул чек, то работаем с ним. В противном случае (например, нет связи с SP) считаем, что чек продажи не определён.

Запрос чека в ЦБД не имеет смысла, поскольку в ККЛ не хранится уникальный номер строки чека продажи. А именно этот параметр требуется для оформления возврата в SP.

После окончания денежных расчётов касса отправляет в SP запрос purchases/returns/create для оформления возврата и списания бонусов. Запрос создаётся только в том случае, если известен чек продажи. Если возврат создан без чека продажи, то невозможно заполнить параметр order_num, и, следовательно, запрос не отправляется.

В запросе указываются изменённые позиции. SP найдёт чек в своей базе по значению параметра order_num.

Ответ из SP заносится в протокол. Касса продолжает работу в обычном режиме.

Список запросов в SailPlay

Общие положения по обмену с SP

https://ru.sailplay.dev/reference/ogranicheniya-api

Время ожидания ответа от SP указывается в карточке компании. Касса отправляет запрос и ждёт ответа указанное время. Если SP не отвечает, то данная ситуация считается ошибкой ‘Сервер лояльности недоступен’. Касса продолжает работу без обмена с SP.

Если в ответе на запрос приходит код 200, то ответ положительный. Любые другие значения кода ответа и отсутствие ответа трактуются как отрицательный ответ. Действия при получении отрицательного ответа зависят от запроса и места вызова.

Запрос	*Место вызова*	*Действие при ошибке*
users/info	Открытие смены	Выставить состояние ‘Связь отсутствует’
	Начало чека при отсутствии связи	Ничего не делать
users/info	Форма идентификации покупателя	Отобразить ответ от информационной строке

send/sms-code	Формирование идентификации покупателя	Отобразить ответ в информационной строке
users/add	Формирование идентификации покупателя	Отобразить ответ в информационной строке
users/update	Формирование идентификации покупателя	Отобразить ответ в информационной строке
Calc	При изменении чека	Выставить состояние ‘связь отсутствует’
	В конце чека	Выставить состояние ‘связь отсутствует’
purchases/new	После оплаты чека	Выставить состояние ‘связь отсутствует’ Поместить чек в список неотправленных чеков
returns/create	После расчёта	Ничего не делать

Тексты запросов и подробные ответы протоколируются в логах кассы.

Действия в случае отсутствия связи с SP

В карточке компании указывается время ожидания ответа на запрос к SP. По истечении указанного времени считается, что связь с SP отсутствует. На экране появляется сообщение для кассира об отсутствии связи с программой лояльности. Соответствующий сигнальный символ в левом верхнем углу экрана загорается красным цветом. Сообщение о потере связи отображается только один раз – при первом неудачном запросе. Для последующих запросов сообщение не выводится. После восстановления связи лампочка загорается зелёным цветом.

Связь с SP проверяется в начале чека запросом *users/info для* телефона 70000000000. Если связь отсутствует, то при работе с данным чеком запросы в SP не посылаются.

Если связь с SP пропала при создании покупки (был успешно выполнен calc, но неудача при *purchases/new*), то чек (параметры запроса) сохраняется в отдельном списке для последующей отправки.

Передача неотправленных чеков выполняется при закрытии смены и в специальном пункте меню.

Информация об участниках (клиентах) программы лояльности

https://ru.sailplay.dev/docs/registracia-klienta

Запись клиента имеет три уникальных идентификатора: внутренний id, номер телефона, адрес почты. Для старых клиентов внутренний id будет заполнен номером дисконтной карты. Чем SP будет заполнять внутренний id для новых клиентов, для нас не имеет значения.

Ввод адреса электронной почты, объединение клиентов будут выполняться только в SP.

Запрос ‘Получить информацию о клиенте’ *users/info*

https://ru.sailplay.dev/reference/users-info

параметр	*описание*
token	Токен

store_department_id	Код из глобальных настроек
origin_user_id	Номер телефона или id дисконтной карты

Ответ:

Параметр	*Описание*
status	Код ответа
id	Идентификатор клиента
email	email
phone	телефон
first_name	Имя
middle_name	Отчество
last_name	Фамилия
birth_date	Дата рождения
origin_user_id	Номер дисконтной карты
points	
confirmed	Подтверждённое количество баллов

Запрос ‘Отправить sms на любой номер’ *send/sms-code*

https://ru.sailplay.dev/reference/sender-messages-send-sms-code

Параметр	*Описание*
token	Токен
store_department_id	код из глобальных настроек
user_phone	Номер телефона
text	Константа: Код подтверждения: $[sms_code]

Ответ:

Параметр	*описание*
status	Код ответа
sms-code	Код подтверждения

Запрос ‘Регистрация клиента’ *users/add*

https://ru.sailplay.dev/reference/users-add

Параметр	*описание*
token	Токен
store_department_id	код из глобальных настроек

target_dep_origin_id	Код магазина без ведущих
user_phone	Номер телефона
first_name	Имя
middle_name	Отчество
last_name	Фамилия
birth_date	Дата рождения

Ответ:

Параметр	*Описание*
status	Код ответа
id	Идентификатор клиента

Запрос ‘Изменить телефон клиента’

users/update

https://ru.sailplay.dev/reference/users-update

Параметр	*Описание*
token	Токен
store_department_id	код из глобальных настроек
new_phone	Номер телефона
user_id	Идентификатор клиента

Идентификатор клиента можно получить запросами users-info или users-add.

Ответ:

Параметр	*Описание*
status	Код ответа

Расчёт скидок

https://ru.sailplay.dev/docs/raschet-akcii-dlya-korzini

Для промежуточных расчётов применяется запрос calc. Непосредственно перед оплатой запрашивается число баллов и отправляется финальный запрос calc для учёта баллов.

https://ru.sailplay.dev/docs/sozdanie-pokupki

Подтверждение покупки в SP выполняется двумя запросами: создание покупки и её подтверждение. Для создания выполняется запрос purchases/new, который регистрирует покупку в базе. Подтверждение выполнит SP самостоятельно.

Запрос ‘Рассчитать скидку’ *marketing-actions/calc*

https://ru.sailplay.dev/docs/raschet-akcii-dlya-korzini

Параметр	*Описание*
token	Токен
store_department_id	код из глобальных настроек
target_dep_origin_id	Код магазина без ведущих
user_phone	Номер телефона

promocode	промокод
discount_points_writeoff	Число баллов для списания
cart	Json-строка с данными о товарах
1,2 и т.д.	Номер позиции
sku	Код товара в Домино
price	Цена
quantity	Количество
discount_type	Тип цены

Пример строки cart:

{"1":{"sku":"5011921150014","price":1600,"quantity":1,”discount_type”:” Regular” }}

discount_type принимает значения:

- Regular – розничная цена
- Sale – распродажная цена
- Red – последняя цена. Если розничная и распродажная цена совпадают.

Если покупатель не идентифицирован, то параметры user_phone и discount_points_writeoff не передаются.

Ответ:

Параметр	*Описание*
status	Статус ответа
cart	
Id	Уникальный ID расчёта чека
positions	
product	
sku	Код товара
price	Цена
new_price	Новая цена
total_discount_points_max_for_user	Максимально возможное количество баллов для списания по данному чеку
num	Номер позиции

Максимальное число баллов для списания по этому чеку SP передаёт именно в теге total_discount_points_max_for_user.

Запрос ‘Создать покупку’

purchases/new

https://ru.sailplay.dev/reference/purchases-new

Параметр	*Описание*
token	Токен
store_department_id	код из глобальных настроек
target_dep_origin_id	Код магазина без ведущих
user_phone	Номер телефона
order_num	Уникальный номер чека
cart_ID	id расчёта чека

id расчёта чека – ссылка на ранее выполненный расчёт по запросу *Рассчитать скидку.*

Order_num потребуется для возврата и он должен быть уникальным. Формируется из трёх параметров: <код магазина>-<номер кассы>-<номер чека в Домино>. Длина: 10 символов - 3 символа-3символа= 18

Ответ:

Параметр	*Описание*
status	Статус ответа

Никакие другие параметры из ответа не рассматриваются, поскольку все расчёты были сделаны раньше в запросе calc. Создание покупки лишь оформляет чек в SP.

Запрос ‘Подтвердить покупку’
confirm

https://ru.sailplay.dev/reference/purchases-confirm

Обычная схема обмена SP предполагает, что в SP должен быть отправлен запрос purchases/confirm для подтверждения только что созданной покупки. Но в рамках данной задачи отправлять запрос на подтверждение не следует. SP в автоматическом режиме, через заданное время подтвердит покупку.

Возврат товаров

Для оформления возврата у покупателя запрашиваются реквизиты чека продажи. По этим реквизитам выполняется поиск чека продажи. Если продажа была произведена на этой кассе, то чек считывается из базы кассы. В противном случае выполняется запрос чека продажи в SP. После выбора позиций для возврата, чек возврата посылается в SP.

Реализован только частичный возврат, полный возврат по чеку (отмена покупки) в данной версии обмена не предусмотрен.

Запрос ‘Создать возврат’
purchases/returns/create/

https://ru.sailplay.dev/reference/purchases-returns-create

Это запрос для частичного возврата товара, когда покупатель отказался от некоторых позиций, новые позиции не добавлялись.

Параметр	*Описание*
token	Токен
store_department_id	код из глобальных настроек

target_dep_origin_id	Код магазина без ведущих
user_phone	Номер телефона
order_num	уникальный номер чека
return_cart	
1, 2 и т.д.	Номер позиции в чеке покупки
quantity	Количество штук возврата
reason	Причина возврата

Уникальный номер чека был передан в SP в запросе ‘Создать покупку’.

Пример заполнения return_cart: "1": {"quantity": 3, "reason": "Брак"},"3": {"quantity": 1, "reason": "Не подошёл размер"}}'

Ответ:

Параметр	*Описание*
status	Статус ответа

Запрос ‘Получить покупку’ *purchases/get*

https://ru.sailplay.dev/reference/purchases-get

Параметр	*Описание*

token	Токен
store_department_id	код из глобальных настроек
order_num	Уникальный номер чека

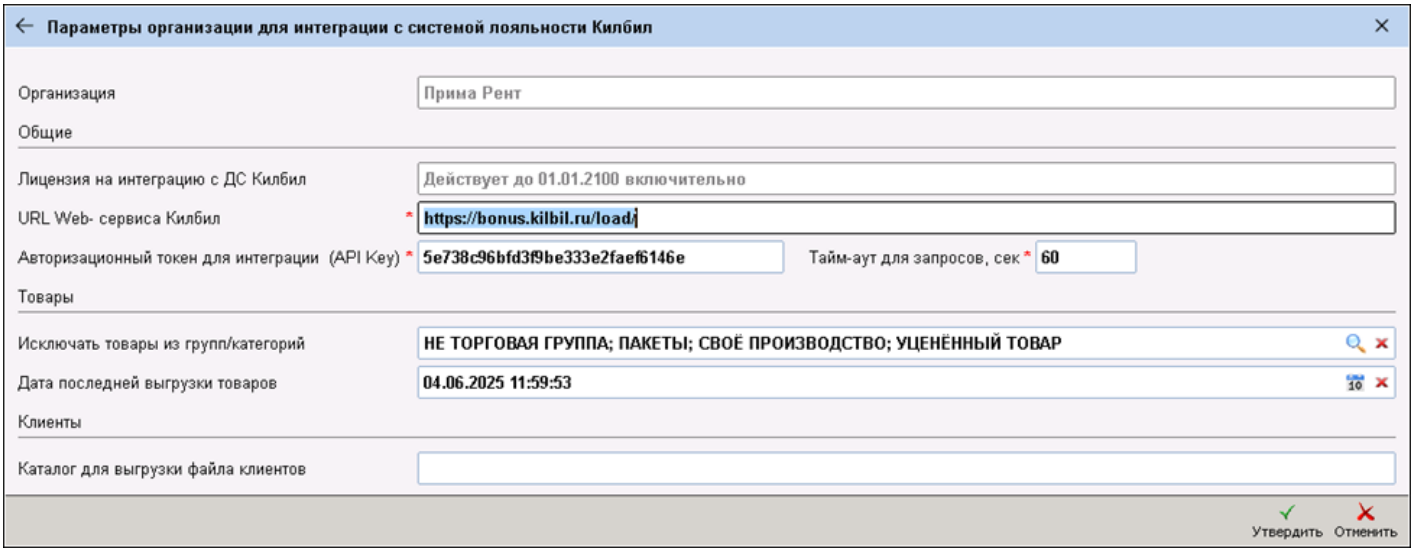
Ответ:

Параметр	*Описание*
status	Статус ответа
cart	
positions	
product	
sku	Код товара
price	Цена
new_price	Новая цена
num	Номер позиции
quantity	количество

Kilbil

Настройка

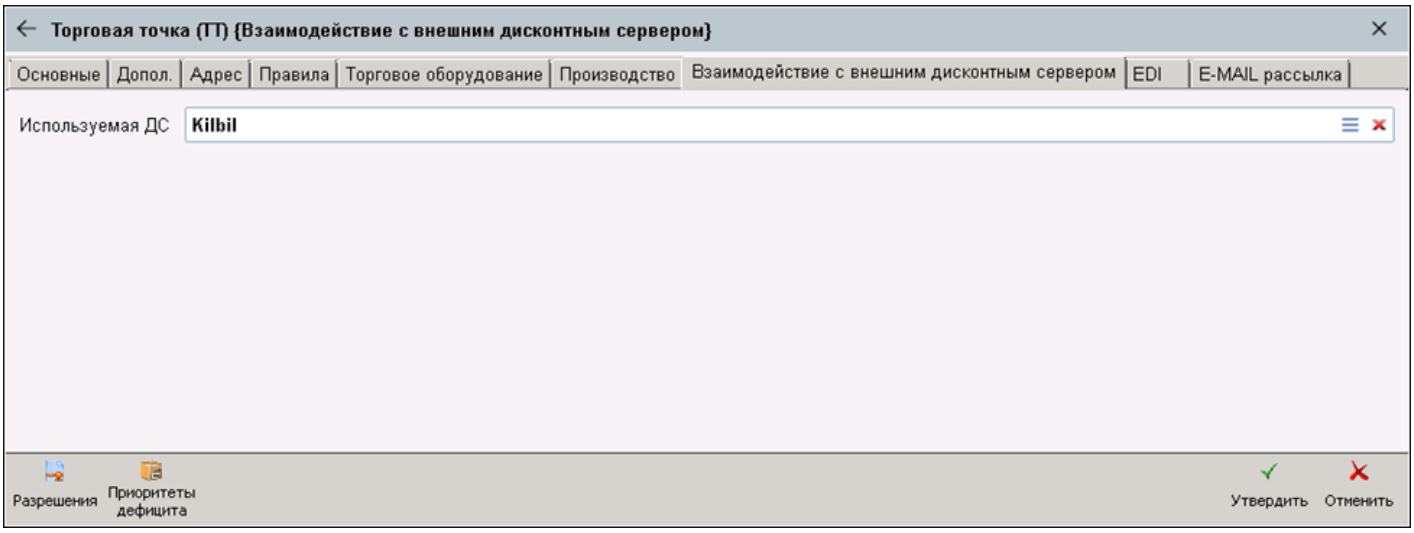
В карточке компании, кнопка "Интеграция с внешними ДС", пункт меню Kilbil, заполняются параметры в разделе "Общие":



- URL Web-сервиса Kilbil
- Авторизационный токен для интеграции (API Key)
- Тайм-аут для запросов, сек

Авторизационный токен допускает выгрузку из базы данных товаров и партнеров (покупателей).

В карточке структурного подразделения (торговой точки) заполняется параметр "Используемая ДС".



Обмен касс магазина с Kilbil включается только в том случае, если параметр заполнен будет.

В карточках точек продаж (POS) заполняются:

- Авторизационный токен для касс (API Key)
- Тайм-аут для запросов в секундах. Это время ожидания ответа от Kilbil на запрос кассы. Если за указанное время Kilbil не ответит, то касса будет обрабатывать эту ситуацию как ошибку обмена (отсутствие связи).

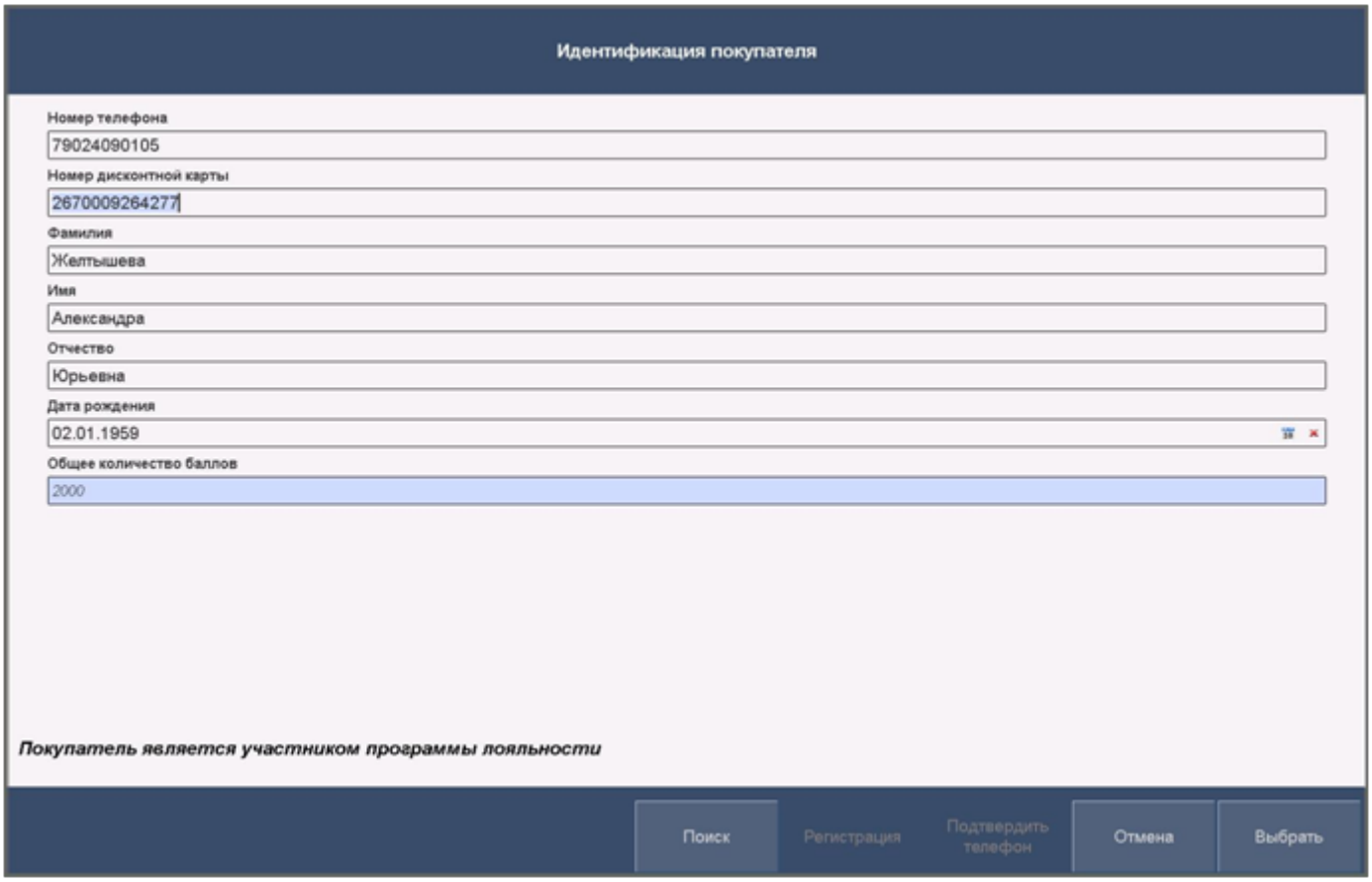
В настройке интерфейса кассира следует привязать к горячим клавишам две команды: ‘*Идентификация покупателя в ДС*’, ‘*Запрос промокода’*. Эти команды используются для запуска соответствующих процессов.

Kilbil

Сценарий идентификации покупателя

Процесс запускается по команде ‘*Идентификация покупателя в ДС*’. По окончании процесса касса получает идентификатор покупателя. Если идентификатор не будет заполнен, то считается, что покупатель не участвует в программе лояльности.

На экране отображается форма с данными покупателя, информационной строкой, кнопками для выбора действий.



Поля:

- Номер телефона (начинается с 7)
- Номер карты – можно сканировать ШК, можно ввести вручную часть номера

- Фамилия
- Имя
- Отчество
- Дата рождения
- Число баллов (для просмотра)
- Строка с сообщением для кассира

Кнопки:

- Поиск
- Регистрация
- Отмена
- Подтвердить телефон
- Выбрать

Вся дальнейшая работа внутри процесса выполняется в данной форме. Форма закрывается только при выходе из процесса. Промежуточные результаты отображаются в полях формы и в информационной строке.

Сначала кассир вводит один из идентификаторов покупателя: номер телефона или номер карты. Номер телефона начинается с цифры 7.

После ввода телефона выполняется запрос в КВ. КВ по номеру телефона ищет в своей БД запись участника программы. Если поиск завершается успешно, то КВ в ответе на запрос возвращает информацию об участнике программы. Касса заполняет остальные поля формы данными из ответа.

Основные сценарии работы кассира

1. Покупатель является участником программы лояльности

Если покупатель говорит, что он является участником программы лояльности и называет номер телефона. Кассир вводит номер и нажимает клавишу 'Enter'. Касса отправляет в КВ запрос. КВ по номеру телефона (или номеру карты) ищет в своей БД запись участника программы. Если поиск завершается успешно, то КВ в ответе на запрос возвращает информацию об участнике программы. Касса заполняет остальные поля формы данными из ответа. Далее проверяется наличие значения в поле Телефон. Если поле не заполнено, то в

информационной строке отображается тест: ‘*Требуется подтверждение телефона*’. Если телефон заполнен, то в информационной строке появляется сообщение ‘*Покупатель является участником программы лояльности*’. Если КВ присылает отрицательный ответ (подходящая запись в БД не обнаружена), то в информационной строке отображается ответ из КВ.

Далее кассир анализирует информацию на экране и действует по одному из следующих сценариев.

- *1.1. Участие подтверждено, телефон имеется. В строке сообщения находится текст ‘Покупатель является участником программы лояльности’*

Кассир нажимает кнопку ‘*Выбрать*’. Форма закрывается. Идентификатор покупателя определён.

- *1.2. Участие подтверждено, не указан телефон. В строке сообщения находится текст ‘Требуется подтверждение телефона’*

Кассир вводит номер телефона и нажимает кнопку ‘*Подтвердить телефон*’.

Сначала касса отправляет запрос для проверки, что данный телефон ещё не зарегистрирован. Если ответ положительный, то в информационной строке отображается текст ‘*Покупатель с таким телефоном уже зарегистрирован*’. Выполняется возврат в форму. Кассир либо вводит другой номер телефона, либо отменяет идентификацию.

Если ответ отрицательный (т.е. в базе КВ нет регистрации данного телефона), то касса отправляет в КВ запрос для подтверждения номера. Если ответ отрицательный, то в информационную строку записывается сообщение об ошибке. Выполняется возврат в форму.

В случае положительного ответа отображается новая форма для запроса кода. В форме одно поле ‘Введите код’ и две кнопки: ‘Проверить’, ‘Отмена’. Покупатель называет код, полученный по sms, кассир вводит этот код и нажимает ‘проверить’. Касса отправляет запрос на проверку полученного кода. КВ возвращает результат проверки.
Если коды совпадают, то касса отправляет в КВ запрос на изменение данных покупателя. Если ответ положительный, то в информационной строке отображается ‘Телефон подтверждён’. Выполняется возврат в основную форму. В информационной строке появляется сообщение ‘Покупатель является участником программы лояльности’.

Если получен отрицательный ответ на запрос изменения данных покупателя, то выполняется возврат в форму, в информационной строке отображается сообщение об ошибке.

Если код, названный покупателем, не совпадают с кодом, полученным по запросу, то в форме для запроса отображается ‘*неверный код’*. Кассир может повторно ввести код и нажать ‘*проверить*’. Если кассир нажимает ‘*Отмена*’, то выполняется возврат в основную форму без подтверждения телефона. В информационной строке остаётся текст ‘ *Требуется подтверждение телефона*’.

- *1.3. Участие не подтверждено. Текст в информационной строке содержит описание ошибки*

Дальнейшие действия либо по ветке 2, либо по ветке 3.

2. Покупатель не является участником программы лояльности и не хочет участвовать

Кассир выбирает ‘*Отмена*’. Форма закрывается, идентификация завершается без заполнения идентификатора покупателя.

3. Покупатель не является участником программы лояльности и хочет зарегистрироваться

Кассир заполняет в форме остальные параметры. Обязательно должны быть заполнены телефон, Имя, Фамилия. Отчество и дата рождения не обязательны к заполнению. Номер карты для регистрации не нужен и будет очищен.

Далее выполняются действия по проверке телефона в базе КВ и его подтверждения. Подробно эти действия описаны в ветке 1.2. Алгоритм действий такой же, но после успешного подтверждения телефона регистрация продолжается. Касса отправляет запрос на регистрацию. Если в ответе будет ошибка, то отображается текст ошибки в информационной строке. Выполняется возврат в основную форму. В случае положительного ответа форма закрывается. Идентификатор покупателя определён.

Запрос промокода

Процесс запускается по команде ‘*Запрос промокода*’. По окончании процесса касса получает название промокода. Если название промокода не заполнено, то считается, что промокода нет.

На экране отображается форма с одним полем ‘Промокод’ и двумя кнопками ‘*Подтвердить*’ и ‘*Отмена*’. Никакие проверки введённой строки не выполняются.

Промокод будет применён в запросах в КВ для расчёта скидок.

Никаких ограничений на доступ к команде нет. Доступность команды не зависит от наличия обмена с КВ или идентификации покупателя.

Промокод на чек может быть только один. Повторный ввод промокода удаляет предыдущий.

Процесс Расчёт скидок

Данный процесс является неотъемлемой частью основных кассовых процессов заполнения и оплаты чека.

Условно процесс делится на четыре этапа:

- Регулярный расчёт скидок по чеку
- Запрос у покупателя количества баллов для оплаты чека
- Финальный расчёт чека
- Создание чека в КВ

Этапы процесса выполняются при следующих условиях:

- включён обмен с КВ (заполнены параметры в карточке точки продаж, имеется связь с КВ)
- магазин является участником программы лояльности (заполнен параметр в карточке магазина)
- чек не является оформлением покупки по интернет-заказу (кассир не запрашивал номер интернет-заказа)

Для запроса баллов и расчёта скидок за баллы дополнительно требуется успешная идентификация покупателя. Если покупатель не идентифицирован как участник программы лояльности, то баллы не запрашиваются.

Регулярный расчёт скидок по чеку

После каждого изменения содержания чека (добавление и удаление товаров, изменение количества) касса отправляет в КВ запрос на расчёт чека. Число баллов в запросе равно 0, чтобы КВ при расчёте скидок не учитывал баллы. В случае положительного ответа от КВ касса разбирает ответ и изменяет фактические цены товаров. Если КВ не ответил или вернул отрицательный ответ, то цены не меняются.

При отсутствии связи с КВ на экране красным цветом загорается соответствующий сигнальный символ. Если связь прервалась после предыдущего успешного запроса, то на экране высвечивается сообщение для кассира ‘*Сервер лояльности недоступен*’. Если на предыдущий запрос также не было ответа, то повторно сообщение не высвечивается.

Запрос баллов

Выполняется только при условии успешной Идентификации покупателя.

В процессе Идентификации КВ вернул число баллов. Эти баллы можно потратить для оплаты чека – на товары может быть рассчитана дополнительная скидка в рублях, равная числу баллов.

После перехода в режим оплаты на экране появляется окно с предложением ввести количество баллов для оплаты чека. По умолчанию в поле находится имеющееся значение баллов. Кассир по согласованию с покупателем может уменьшить это число. В форме имеются две кнопки: ‘*Подтвердить*’ и ‘*Отмена*’. Если кассир нажал ‘*Подтвердить*’, то число баллов для оплаты принимается равным введённому значению. Выбор кнопки ‘*Отмена*’ означает отказ от списания баллов.

Финальный расчёт чека

Выполняется только при условии успешной Идентификации покупателя, после запроса баллов. Касса отправляет в КВ финальный запрос для расчёта скидок. В запросе должен быть заполнен тег с числом балов, чтобы КВ при расчёте скидок учитывал баллы. Значение берётся из ответа на предыдущий запрос расчета чека.

В случае положительного ответа от КВ касса разбирает ответ и изменяет фактические цены товаров. Если КВ не ответил или вернул отрицательный ответ, то касса ничего не делает.

Если покупатель отказывается от оплаты и возвращается в редактирование чека, то следует сбросить расчёт с учётом баллов. Для этого касса отправляет в КВ запрос на расчёт чека с нулевым количеством баллов.

Подтверждение оплаты чека

Выполняется после полного оформления чека, когда покупатель оплатил покупку и чек отправлен на ФР.

Касса отправляет в КВ запрос для подтверждения покупки.

Если КВ вернул ошибку, то повторяем расчет чека и ещё раз подтверждение покупки. Если опять ошибка, то чек сохраняется в списке неотправленных чеков.

Если связь с КВ отсутствует, то чек также сохраняется в списке неотправленных чеков.

Список неотправленных чеков нужен только для тех чеков, по которым был выполнен расчёт скидок, покупатель оплатил чек, но информация о покупке в КВ не была передана. Т.е. КВ расчёты выполнил, но запрос на создание покупки не получил.

Kilbil

Процесс Возврат товара

Покупатель оформляет возврат товаров в кассе. Предполагается, что это отказ от части позиций, без добавления новых позиций.

Кассир вводит реквизиты чека продажи (фискальный номер, дата, номер смены). Поиск чека сначала выполняется в базе кассы. Если чек не обнаружен, то выполняется поиск в ЦБД. Цель поиска - определить реквизиты чека продажи в Домино для формирования номера этого чека в КВ.

При создании чека в КВ номер чека был создан по следующему шаблону: <Код структурного подразделения> - <код кассы> - <номер смены> - <дата чека> - <номер чека>.

Выполняется запрос покупки в КВ. Если КВ вернул чек, то работаем с ним. В противном случае (например, нет связи с КВ) считаем, что чек продажи не определён. Для оформления возврата нужны номера позиций в чеке. В запросе указываются изменённые позиции.

После окончания денежных расчётов касса отправляет в КВ запрос для оформления возврата и списания бонусов. Запрос создаётся только в том случае, если известен чек продажи. Если возврат создан без чека продажи, то невозможно заполнить идентификатор чека в КВ, и, следовательно, запрос не отправляется.

Ответ из КВ заносится в протокол. Касса продолжает работу в обычном режиме.

Список запросов в KilBil

Общие положения по обмену с КВ

https://ru.sailplay.dev/reference/ogranicheniya-api

Запуск запроса: https://bonus.kilbil.ru/load/{название_функции}?h={ключ}

Тело запроса должно соответствовать формату JSON.

Время ожидания ответа от КВ указывается в карточке компании. Касса отправляет запрос и ждёт ответа указанное время. Если КВ не отвечает, то данная ситуация считается ошибкой ‘Сервер лояльности недоступен’. Касса продолжает работу без обмена с КВ.

Если в ответе на запрос приходит код 200, то ответ положительный. Любые другие значения кода ответа и отсутствие ответа трактуются как отрицательный ответ.

Действия при получении отрицательного ответа зависят от запроса и места вызова.

Запрос	*Место вызова*	*Действие при ошибке*
searchklient	Открытие смены	Выставить состояние ‘Связь отсутствует’
	Начало чека при отсутствии связи	Ничего не делать

searchclient	Формат идентификации покупателя	Отобразить ответ идентификации в информационной строке
askconfirmphone *checkconfirmphone*	Формат идентификации покупателя	Отобразить ответ идентификации в информационной строке
addclient	Формат идентификации покупателя	Отобразить ответ идентификации в информационной строке
processchange	При изменении чека	Выставить состояние ‘связь отсутствует’
	В конце чека	Выставить состояние ‘связь отсутствует’
confirmsale	После оплаты чека	Выставить состояние ‘связь отсутствует’ Поместить чек в список неотправленных чеков
getdatatakingbymove	Запрос итогов продажи	Выставить состояние ‘связь отсутствует’

Тексты запросов и подробные ответы протоколируются в логах кассы.

Действия в случае отсутствия связи с КВ

В карточке компании указывается время ожидания ответа на запрос к КВ. По истечении указанного времени считается, что связь с КВ отсутствует. На экране появляется сообщение для кассира об отсутствии связи с программой лояльности. Соответствующий сигнальный символ в левом верхнем углу экрана загорается красным цветом. Сообщение о потере связи отображается только один раз – при первом неудачном запросе. Для последующих запросов сообщение не выводится. После восстановления связи лампочка загорается зеленым цветом.

Связь с КВ проверяется в начале чека запросом *searchklient для* телефона 70000000000. Если связь отсутствует, то при работе с данным чеком запросы в КВ не посылаются.

Если связь с КВ пропала при создании покупки (был успешно выполнен *processsale*, но неудача при *confirmsale*), то чек (параметры запроса) сохраняется в отдельном списке для последующей отправки.

Передача неотправленных чеков выполняется при закрытии смены и в специальном пункте меню.

Информация об участниках (клиентах) программы лояльности

Запись клиента имеет два уникальных идентификатора: номер карты, номер телефона. Для старых клиентов номер карты будет заполнен, а номер телефона может быть пустым.

В Домино требуется проверять уникальность идентификаторов при создании нового клиента. Ограничимся проверкой только номера телефона. Поскольку остальные уникальные идентификаторы при регистрации нового клиента можно не заполнять, то так и будем поступать.

Ввод адреса электронной почты, объединение клиентов будут выполняться только в личном кабинете КВ. https://help.kilbil.ru/93-1075-614--kak-dobavit-klienta-cherez-lichnyj-kabinet/

Запрос ‘Получить информацию о клиенте’ *searchclient*

https://bonus.kilbil.ru/load/searchclient

Запрос:

Параметр	*Описание*
search_mode	0 – поиск по № телефона 2 – поиск по № карты
search_value	значение, по которому производится поиск, если по номеру телефона, это должно быть 11 символов первая 7-ка

Ответ:

Параметр	*Описание*
result_code	код результата, где 0 - успешно

client_id	id клиента, null если не найден
bonus_balance	количество бонусов у клиента
card_code	номер карты, null если нет
phone	номер телефона клиента, null если не заполнено
birth_date	дата рождения клиента, null если не заполнено
first_name	имя клиента, null если не заполнено
middle_name	отчество клиента, null если не заполнено

last_name	фамилия клиента, null если не заполнено
max_bill_bonus_out	максимальная сумма списания на текущий чек без учета баланса клиента, параметр нужен для дальнейшего использования в processsale

Запрос ‘Отправить код подтверждения’ *askconfirmphone*

https://bonus.kilbil.ru/load/askconfirmphone

Запрос:

Параметр	*Описание*
client_id	id клиента
phone	номер телефона клиента 11 символов, первая 7

sms_type	тип шаблона сообщения 0 - сообщение с пин-кодом подтверждения для регистрации клиента 1 - сообщение с пин-кодом, например для списания бонусов

Ответ:

Параметр	*Описание*
success	код выполнения
result_code	код результата, где 0 - успешно

Запрос ‘Проверить код подтверждения’ *checkconfirmphone*

https://bonus.kilbil.ru/load/checkconfirmphonecode

Запрос:

Параметр	*Описание*

client_id	id клиента
phone	номер телефона клиента 11 символов, первая 7
code	код подтверждения

Ответ:

Параметр	*Описание*
success	код выполнения
result_code	код результата, 0 - успешно

Запрос ‘Добавление (обновление) клиента’ *addclient*

https://bonus.kilbil.ru/load/addclient

Запрос:

Параметр	*Описание*
client_id	id клиента

phone	номер телефона клиента 11 символов, первая 7, клиент не будет активирован без номера телефона

Ответ:

Параметр	*Описание*
success	код выполнения
client_id	id клиента
result_code	код результата, где 0 - успешно

Расчёт скидок

Для промежуточных расчётов применяется запрос processsale. Непосредственно перед оплатой следует запросить число баллов и отправить финальный запрос processsale для учёта баллов.

Подтверждение покупки в КВ выполняется отдельным запросом.

Запрос ‘Рассчитать документ’ *processsale*

https://bonus.kilbil.ru/load/processsale

Параметр	*Описание*
client_id	id клиента, null – если покупатель не идентифицирован
type	тип операции: 0 – продажа, 1 - возврат
bonus_out	количество списанных бонусов в операции
max_bonus_out	максимально возможная сумма списания бонусов без учета баланса клиента (передается значение выходного параметра max_bill_bonus_out функции searchclient)
good_data	Массив товаров

code	Код товара
vendor_code	Артикул
name	Наименование товара
price	цена
quantity	количество
total	Сумма за позицию
minPrice	Минимальная цена
discounted_price	Цена за позицию, равна price
discounted_total	Сумма за позицию, равна total
move_id	Номер чека
doc_open_dt	Дата чека

return_receipt_number	номер чека продажи, по которому производится возврат (передаётся только для возвратов)
return_date	Дата чека продажи (передаётся только для возвратов)
promo_codes	Список промокодов
coupons	Массив купонов
coupon	промокод

Move_id потребуется для возврата и он должен быть уникальным. Заполняем из следующих параметров: <Код структурного подразделения> - <код кассы> - <номер смены> - <дата чека> - <номер чека>.

Поскольку применяются только скидки через систему лояльности, то price=discounted_price, Total= discounted_total.

Промокод может быть только один. Кассир его ввёл в процессе ‘запрос промокода’.

Ответ:

Параметр	*Описание*

result_code	код результата, code 0 – успешно
bill_data	информация по расчету документа
items	Массив товаров
code	Код товара
discounted_price	Новая цена
price	Цена
bonus_out	Количество списанных бонусов на позицию
bonus_out	Количество списанных бонусов на документ

Запрос ‘Подтвердить документ’ *confirmsale*

https://bonus.kilbil.ru/load/confirmsale

Подтверждение документа после processsale, после этого в системе лояльности чек считается подтвержденным и участвует в отчетах. Выполняются все начисления и списания бонусов.

Параметр	*Описание*

move_id	Номер чека
doc_open_dt	Дата чека
doc_dt	Дата закрытия чека

Ответ:

Параметр	*Описание*
result_code	код результата, где 0 – успешно

Возврат товаров

Для создания чека возврата применяется запрос ‘Рассчитать документ’ (processsale с type=1).

Предварительно в базе KB запрашивается чек продажи.

Запрос ‘Получить документ’ getdetailingbymove

https://bonus.kilbil.ru/load/getdetailingbymove

Параметр	*Описание*
move_id	Номер чека

Ответ:

Параметр	*Описание*

result_code	код результата, code 0 – успешно
move_info	общая информация по чеку
oper_type	Тип операции: 0 –продажа
move_detail	Массив товаров
code	Код товара
price	Цена
total	Сумма по позиции
quantity	количество

Выгрузка справочника товаров в KilBil

https://docs.kilbil.ru/#tag/Nomenklatura/operation/goods

Предусмотрены два режима выгрузки товаров в КВ:

1. Полная выгрузка
2. Инкрементальная выгрузка

Полная выгрузка запускается вручную из меню. Выгружаются все товары в КВ.

Товары выгружаются порциями по 2000 товаров.

Инкрементальная выгрузка обрабатывает только изменения товаров с момента предыдущей выгрузки. Запускается либо вручную из меню, либо автоматически планировщиком.

Совместно с карточками товаров выгружаются группы и категории, 4 уровня классификатора.

Запрос:

Параметр	*Описание*
partial	0 – полная, 1 – инкрементальная

is_group	True – для групп и категорий False – для товаров
delete_mark	True в случае инкрементальной выгрузки
code	Код товара в Домино или код группы/категории
name	Наименование товара или наименование группы/категории
Inside	Следующий уровень группы/категории внутри группы или товар внутри категории. Включает параметры is_group, delete_mark, code, name, inside

Ответ:

Параметр	*Описание*
Success	1 – успешно
result_code	0 - успешно

> Пример:
> *{"is_group": true,*
> *"delete_mark": false,*
> *"code": "1",*
> *"name": "Название группы",*
> *"inside": []*
> *},*
> *{*
> *"is_group": true,*
> *"delete_mark": false,*
> *"code": "5",*
> *"name": "Название группы 3",*
> *"inside": [*
> *{*
> *"is_group": false,*

```
        "delete_mark": false,
        "code": "706",
        "name": "Наименование товара"
         },
        {
        "is_group": false,
         "delete_mark": false,
          "code": "610",
          "name": "название товара"
          }
    ]
}
```