

Экспорт в retailCRM

retailCRM — это специализированная CRM-система для интернет-магазинов, позволяет обрабатывать заказы, проводить аналитику, управлять товарами и др.

Для работы с API этой системы можно воспользоваться следующим решением.

1. Переносим файлы решения на сервер.
2. Задаем настройки в конфигурационном файле:

```
<?php

return array(
    'url' => 'https://your-shop.retailcrm.ru',
    'api_key' => 'your_api_key',
    'secret_sequence' => 'your_secret_sequence',
    'manager_id' => '1',
    'subscription_order_status' => 'prd',
);
```

3. Пользуемся.

На данный в решение реализованы выгрузка заказа и подписки (решение [«Подписка на отсущствующие товары»](#)), а также работа с событиями.

Выгрузка заказа

Для выгрузки заказа в retailCRM можно использовать следующий наблюдатель:

```
<?php

defined('HOSTCMS') || exit('HostCMS: access denied.');
```

```
/**
 * Наблюдатель для экспорта заказа в retailCRM.
 */
class Shop_Payment_System_Handler_Observer_Retailcrm
{
    /**
     * Регистрирует наблюдатели.
     *
     * @param array $aShops
     * @return void
     */
    static public function attach()
    {
```

```

Core_Event::attach('Shop_Payment_System_Handler.onAfterProcessOrder',
array(__CLASS__, 'exportToRetailCrm'));
}

/**
 * @param Shop_Payment_System_Handler $oShopPaymentSystemHandler
 * @return void
 */
static public function exportToRetailCrm(Shop_Payment_System_Handler
$oShopPaymentSystemHandler)
{
    if (CURRENT_SITE != 1)
    {
        return;
    }

    // Создаем заказ
    $oKadRetailcrmClient = Kad_Retailcrm_Client::instance();
    $oKadRetailcrmClient->createOrder($oShopPaymentSystemHandler->getShopOrder()
);
}
}

```

Выгрузка подписки

Для работы наблюдателя потребуется добавить событие Shop_Item_Subscription_Controller.onAfterAddSubscription к контроллеру Shop_Item_Subscription_Controller, а также поле phone для выгрузки номера телефона.

```

<?php

defined('HOSTCMS') || exit('HostCMS: access denied.');
```

/**
 * Наблюдатель для выгрузки подписки в CRM.
 *
 * @author Maxim Zasorin, KAD Systems (©) 2017
 * @date 14-12-2017
 */

```

class Shop_Item_Subscription_Controller_Observer_DispatchToRetailCrm
{
    /**
     * Регистрирует наблюдатель.
     *
     * @return void
     */
    static public function attach()
    {

```

```

Core_Event::attach('Shop_Item_Subscription_Controller.onAfterAddSubscription', array(__CLASS__, 'addPhoneNumber'));
Core_Event::attach('Shop_Item_Subscription_Controller.onAfterAddSubscription', array(__CLASS__, 'dispatchToRetailCrm'));
}

/**
 * Деактивирует наблюдатель.
 *
 * @return void
 */
static public function detach()
{
Core_Event::detach('Shop_Item_Subscription_Controller.onAfterAddSubscription', array(__CLASS__, 'addPhoneNumber'));
Core_Event::detach('Shop_Item_Subscription_Controller.onAfterAddSubscription', array(__CLASS__, 'dispatchToRetailCrm'));
}

/**
 * Сохраняем номер телефона.
 *
 * @param Shop_Item_Subscription_Controller $oShopItemSubscriptionController
 * @param array $aArgs
 * @return void
 */
static public function addPhoneNumber(Shop_Item_Subscription_Controller $oShopItemSubscriptionController, array $aArgs)
{
    list($oShopItemSubscription) = $aArgs;

    $oShopItemSubscription->phone =
trim(strval(Core_Array::getRequest('phone')));
    $oShopItemSubscription->save();
}

/**
 * Выгружаем подписку в CRM.
 *
 * @param Shop_Item_Subscription_Controller $oShopItemSubscriptionController
 * @param array $aArgs
 * @return void
 */
static public function
dispatchToRetailCrm(Shop_Item_Subscription_Controller $oShopItemSubscriptionController, array $aArgs)
{
    list($oShopItemSubscription) = $aArgs;

```

```
        $oKadRetailcrmClient = Kad_Retailcrm_Client::instance();
        $oKadRetailcrmClient->createOrderFromSubscription($oShopItemSubscription);
    }
}
```

Работа с событиями

Для работы с событиями (называются триггерами), которые отсылает retailCRM реализован специальный класс, использовать его необходимо следующим образом:

```
<?php

/**
 * Событие изменения заказа в retailCRM.
 *
 * @author Maxim Zasorin, KAD Systems (©) 2017
 * @date 07-12-2017
 */

// Обработка запроса
$oKadRetailcrmWebhook =
Kad_Retailcrm_Webhook::create(file_get_contents('php://input'));

if ($oKadRetailcrmWebhook)
{
    $oData = $oKadRetailcrmWebhook->getData();

    // Выполняем какие-то действия с пришедшими данными $oData, здесь уже точно
    известно, что они пришли от retailCRM
}
else
{
    Core_Page::instance()->error403();
}
```

Страницу создал Максим Засорин 28.12.17 в 09:57