

Документация для разработчиков

Документация Rest API

Введение

API позволяет сторонним системам осуществлять покупку уровня доступа к вашему сервису. Это API обеспечивает возможность создания заказов через HTTP-запросы с использованием JSON-формата.

Описание:

Method: **POST**

URL: *https://meberlux_site_ulr/wp-json/mbl/v1/order/*

Content-type: **application/json**

Authorization: **Basic base64(user:password)**

Body:

```
{
  product_id: int | int[] required
  email: string required
  first_name: string optional
  last_name: string optional
  patronymic: string optional
  phone: string optional
  custom1: string optional
  custom2: string optional
  custom3: string optional
  custom4: string optional
  custom5: string optional
  custom6: string optional
  custom7: string optional
  custom8: string optional
  custom9: string optional
  custom10: string optional
  custom1textarea: string optional
}
```

RESPONSE:

status: 200

```
json
{
    "order": int
}
```

user - это логин пользователя (администратора), у которого создан пароль приложения на первом шаге, например admin, password - пароль приложения, например UOHV QJBC y5RJ Tcka j76F sOhK.

Поля custom1-custom10, custom1textarea - из MEMBERLUX - настройки - Страницы - Регистрация.

Примеры:

1. PHP

```
<?php
$url = 'http://your-wordpress-site.com/wp-json/mb1/v1/order/';

$data = array(
    'product_id' => [1723, 2523],
    'email' => 'example@gmail.com',
    'first_name' => 'Алексей',
    'last_name' => 'Логинов',
    'patronymic' => 'Борисович',
    'phone' => '79005231265',
);
$data_json = json_encode($data);

$ch = curl_init($url);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');
curl_setopt($ch, CURLOPT_POSTFIELDS, $data_json);
curl_setopt($ch, CURLOPT_HTTPHEADER, array(
    'Content-Type: application/json',
    'Authorization: Basic ' . base64_encode("username:password")
));
```

```
$response = curl_exec($ch);

if(curl_errno($ch)) {
    echo 'Ошибка cURL: ' . curl_error($ch);
}

curl_close($ch);

echo $response;
```

2. CURL

```
curl -X POST \
  https://etis.top/wp-json/mbl/v1/order/ \
  -H 'Authorization: Basic
b2Jha2FuYWM1NDVfZzR3MDZiZjg6aFFKMSBEM3EzIHdnSXogYXNk
SyBod3JoIGhMSXI=' \
  -H 'Content-Type: application/json' \
  -d '{
    "product_id": [1723, 2523],
    "email": "example@gmail.com",
    "first_name": "Алексей",
    "last_name": "ЛОГИНОВ",
    "patronymic": "Борисович",
    "phone": "79005231265"
  }'
```

Где

b2Jha2FuYWM1NDVfZzR3MDZiZjg6aFFKMSBEM3EzIHdnSXogYXNk
SyBod3JoIGhMSXI= это закодированная строка в base64
username:password

Webhook API

Введение

Webhook API предоставляет возможность вашему приложению автоматически получать уведомления о событиях, происходящих в **Memberlux**. Эти уведомления представляют собой HTTP POST-запросы, отправляемые на указанный вами URL-адрес при возникновении определенных событий.

Как работает **Webhook API**

Webhook API отправляет **POST**-запросы с данными о событии на ваш **URL**-адрес обработчика. Этот запрос содержит информацию о типе события и соответствующих данных.

URL-адрес обработчика

Вы должны предоставить **URL**-адрес обработчика (эндпоинт), на который будут отправляться **POST**-запросы с уведомлениями о событиях. Ваш сервер должен быть настроен для обработки этих запросов и выполнения соответствующей обработки данных.

Типы событий и формат данных

Webhook API поддерживает следующие типы событий:

1. **free_registration_form** - Форма бесплатной регистрации получение ключа
2. **bulk_operations_reg** - Массовые операции (регистрации) получение ключа
3. **bulk_operations_add** - Массовые операции (добавление) получение ключа
4. **auto_registration** - Авторегистрация получение ключа
5. **activation_page** - Добавление ключа на странице активации
6. **profile_page_self** - Добавление ключа в профиле
7. **profile_page_admin** - Добавление доступа администратором
8. **after_auto_training_passed** - После прохождения автотренинга получение ключа из настроек автотренинга.

Каждое событие содержит различные данные, включая тип события и дополнительные параметры, связанные с этим событием. Например, событие **profile_page_admin** содержит информацию о добавлении доступа администратором, такую как адрес электронной почты, и информацию о ключе доступа.

Параметр	Описание
----------	----------

type	Тип события. Может принимать одно из значений: free_registration_form, bulk_operations_reg и т.д.
email	Адрес электронной почты пользователя, связанного с событием
first_name	Имя пользователя, связанного с событием
last_name	Фамилия пользователя, связанного с событием
patronymic	Отчество пользователя, связанного с событием
phone	Телефон пользователя, связанного с событием
custom1	Дополнительное поле 1
custom2	Дополнительное поле 2
custom3	Дополнительное поле 3
custom4	Дополнительное поле 4
custom5	Дополнительное поле 5
custom6	Дополнительное поле 6
custom7	Дополнительное поле 7
custom8	Дополнительное поле 8
custom9	Дополнительное поле 9
custom10	Дополнительное поле 10
custom1textarea	Дополнительное текстовое поле
key	Данными о ключе доступа.
key[term_id]	Идентификатор уровня доступа.
key[name]	Название уровня доступа.
key[is_unlimited]	Флаг, указывающий, что доступ неограничен (1 - да, 0 - нет).
key[duration]	Продолжительность доступа целое число. Если is_unlimited равен 1, то значение null
key[units]	Единицы измерения продолжительности доступа: months или days. Если is_unlimited равен 1, то значение null
key[date_start]	Дата начала доступа dd-mm-yyuu.

key[date_end]	Дата окончания доступа dd-mm-yyuu. Если is_unlimited равен 1, то значение null
---------------	--

Обработка уведомлений

Ваш URL-адрес обработчика должен быть настроен для приема и обработки уведомлений, отправленных Webhook API. При получении уведомления ваш сервер должен ответить со статусом 200 в течение 5 секунд. Ответ может быть пустым или содержать любое содержимое.

Это основные принципы работы Webhook API. Пользуйтесь этим функционалом для автоматизации ваших процессов и получения уведомлений о важных событиях в вашей системе.