MULTIGURA

MULTIGURA Promotion API

Руководство по интеграции для партнёров

КАК ЭТО РАБОТАЕТ

1. Участник регистрируется на вашем сайте или в приложении.
2. Ваш сервер отправляет запрос `POST /account/set`.
3. MULTIGURA создаёт аккаунт, выдаёт лицензию и отправляет письмо для активации.

Базовый адрес
https://multigura.com/wp-json/promotion

Онлайн-версия
multigura.com/ru/api/

Дата
Ноябрь 2026

1. Обзор

Promotion API позволяет нашим партнёрам дарить своим клиентам, студентам или участникам сообщества бесплатную годовую лицензию MULTIGURA. Он создан для университетов, хостинг-провайдеров, агентств и сообществ, которые проводят акцию или кампанию.

Регистрация полностью автоматическая: участник регистрируется на вашем сайте или в приложении, а MULTIGURA сразу создаёт для него аккаунт и выдаёт лицензию. Письмо для активации отправляется пользователю на его языке.

Каждый аккаунт привязан к вашему промокоду. Когда ваши пользователи позже покупают или продлевают лицензию, вы получаете долю их платежей на условиях договора, а для них можно установить и скидку.

Чтобы стать партнёром, свяжитесь с нами. После заключения договора вы получите данные интеграции: для интеграции достаточно двух запросов.

2. Начало работы

После заключения договора с MULTIGURA вы получаете **id** и **secret** своей интеграции. Храните secret только на своём сервере: он никогда не должен попадать в браузер или мобильное приложение.

Базовый адрес: **https://multigura.com/wp-json/promotion**

Каждый запрос содержит **hash**: SHA-256 от значений запроса и MD5 от secret. Он подтверждает, что запрос отправлен вами. Запрос с неверным hash или с id, который не является активной интеграцией, получает ответ HTTP 401.

ПРИМЕР

```
// Check account (GET /account/get)
$hash = hash('sha256', $id . $email . md5($secret));

// Create account (POST /account/set)
$hash = hash('sha256', $id . $user . $email . md5($secret));
```

3. Проверка аккаунта: GET /account/get

Проверяет, зарегистрирован ли email в MULTIGURA.

Параметр	Тип	Описание
`id`	int	Идентификатор интеграции.
`email`	string	Email пользователя.
`hash`	string	SHA-256 от id, email и MD5 от secret.

Если email свободен, ответ содержит status 0 и пустой data. Если email зарегистрирован, status равен 1: data содержит аккаунт, только если пользователь зарегистрирован через вашу интеграцию, иначе data пуст.

Поля company_id и company_name появляются в data, если они заполнены. Дата регистрации (registered) указана в часовом поясе Asia/Tbilisi (UTC+4).

ПРИМЕР

```php
$id     = 1;
$secret = 'YOUR_SECRET';
$email  = 'test@mail.com';

$url = 'https://multigura.com/wp-json/promotion/account/get?' . http_build_query([
    'id'    => $id,
    'email' => $email,
    'hash'  => hash('sha256', $id . $email . md5($secret))
]);

$response = json_decode(file_get_contents($url), true);

// Response
{
    "status": 1,
    "description": "The email already exists",
    "data": {
        "id": 26,
        "login": "test",
        "email": "test@mail.com",
        "registered": "2025-07-06 00:00:00",
        "active": 1,
        "promotion_code_id": 100,
        "promotion_code": "CODE"
    }
}
```

4. Создание аккаунта: POST /account/set

Создаёт аккаунт с бесплатной годовой лицензией. Тело запроса можно отправить как данные формы или как JSON.

Параметр	Тип	Описание
`id`	int	Идентификатор интеграции.
`user`	string	Желаемый логин. Он переводится в нижний регистр, и в нём остаются только латинские буквы и цифры. Если логин начинается с цифры, в начало добавляется user, а если он занят, в конец добавляется номер, поэтому логин в ответе может отличаться.
`email`	string	Email пользователя.
`first_name`	string	Необязательный. Имя.
`last_name`	string	Необязательный. Фамилия.
`company_id`	string	Необязательный. Идентификатор компании.
`company_name`	string	Необязательный. Название компании.
`lang`	string	Необязательный. Язык аккаунта и приветственного письма: en, ge или ru. По умолчанию en.
`hash`	string	SHA-256 от id, user, email и MD5 от secret. Значение user используйте точно в том виде, в каком отправляете.

Если email свободен, MULTIGURA создаёт аккаунт и его лицензию, а если у вашей интеграции есть промокод, привязывает его к аккаунту. Пользователь получает письмо на выбранном языке со ссылкой для установки пароля. Ответ содержит status 0, а data содержит новый аккаунт.

Если email уже зарегистрирован, ничего не создаётся: ответ такой же, как при проверке аккаунта.

ПРИМЕР

```php
$id     = 1;
$secret = 'YOUR_SECRET';

$data = [
    'id'         => $id,
    'user'       => 'test',
    'email'      => 'test@mail.com',
    'first_name' => 'John',
    'last_name'  => 'Smith',
    'lang'       => 'en'
];

$data['hash'] = hash('sha256', $data['id'] . $data['user'] . $data['email'] . md5($secret));

$curl = curl_init('https://multigura.com/wp-json/promotion/account/set');

curl_setopt_array($curl, [
    CURLOPT_POST           => true,
    CURLOPT_POSTFIELDS     => http_build_query($data),
    CURLOPT_RETURNTRANSFER => true
]);

$response = json_decode(curl_exec($curl), true);

// Response
{
    "status": 0,
    "description": "Done",
    "data": {
        "id": 26,
        "login": "test",
        "email": "test@mail.com",
        "registered": "2025-07-06 00:00:00",
        "active": 1,
        "promotion_code_id": 100,
        "promotion_code": "CODE"
    }
}
```

5. Статусы и ошибки

Каждый ответ с HTTP 200 содержит status:

Статус	Описание	Значение
0	Done	Проверка аккаунта: email свободен. Создание аккаунта: аккаунт создан.
1	The email already exists	Email зарегистрирован. Ничего не создаётся.
99	Unknown system error	Запрос не удалось выполнить. Повторите попытку позже, а если ошибка повторяется, свяжитесь с нами.

Запрос, который невозможно обработать, вместо этого получает ошибку HTTP в стандартном формате WordPress:

HTTP	Код	Причина
400	rest_missing_callback_param	Отсутствует обязательный параметр.
400	rest_invalid_param	Неверный параметр, например email.
401	rest_forbidden	Неверный hash, или id не является активной интеграцией.

ПРИМЕР

```
// HTTP 400
{
    "code": "rest_invalid_param",
    "message": "Invalid parameter(s): email",
    "data": {
        "status": 400,
        "params": {
            "email": "Invalid parameter."
        }
    }
}

// HTTP 401
{
    "code": "rest_forbidden",
    "message": "Sorry, you are not allowed to do that.",
    "data": {
        "status": 401
    }
}
```