MULTIGURA

MULTIGURA Promotion API

ინტეგრაციის სახელმძღვანელო პარტნიორებისთვის

როგორ მუშაობს

1. მონაწილე თქვენს საიტზე ან აპლიკაციაში რეგისტრირდება.
2. თქვენი სერვერი აგზავნის `POST /account/set` მოთხოვნას.
3. MULTIGURA უქმნის ანგარიშს, ანიჭებს ლიცენზიას და უგზავნის აქტივაციის წერილს.

ძირითადი მისამართი
https://multigura.com/wp-json/promotion

ონლაინ ვერსია
multigura.com/ge/api/

თარიღი
ნოემბერი, 2026

1. მიმოხილვა

Promotion API ჩვენს პარტნიორებს საშუალებას აძლევს, თავიანთ მომხმარებლებს, სტუდენტებს ან საზოგადოების წევრებს MULTIGURA-ს უფასო ერთწლიანი ლიცენზია აჩუქონ. ის შექმნილია უნივერსიტეტებისთვის, ჰოსტინგ პროვაიდერებისთვის, სააგენტოებისა და საზოგადოებებისთვის, რომლებიც აქციას ან კამპანიას ატარებენ.

რეგისტრაცია სრულად ავტომატურია: მონაწილე თქვენს საიტზე ან აპლიკაციაში რეგისტრირდება, MULTIGURA კი მაშინვე უქმნის ანგარიშს და ანიჭებს ლიცენზიას. აქტივაციის წერილი მომხმარებელს თავის ენაზე ეგზავნება.

ყოველი ანგარიში თქვენს პრომო კოდზეა მიბმული. როცა თქვენი მომხმარებლები შემდგომში ლიცენზიას შეიძენენ ან განაახლებენ, ხელშეკრულების პირობებით მიიღებთ მათი გადახდების წილს, მათთვის კი შესაძლებელია ფასდაკლების დაწესებაც.

პარტნიორობისთვის დაგვიკავშირდით. ხელშეკრულების გაფორმების შემდეგ მიიღებთ ინტეგრაციის მონაცემებს: ინტეგრაციისთვის ორი მოთხოვნა საკმარისია.

2. დაწყება

MULTIGURA-სთან ხელშეკრულების გაფორმების შემდეგ მიიღებთ თქვენი ინტეგრაციის **id**-სა და **secret**-ს. საიდუმლო გასაღები (secret) მხოლოდ თქვენს სერვერზე შეინახეთ: ის არასდროს უნდა მოხვდეს ბრაუზერში ან მობილურ აპლიკაციაში.

ძირითადი მისამართი: **https://multigura.com/wp-json/promotion**

ყოველ მოთხოვნას ახლავს **hash**: მოთხოვნის მნიშვნელობებისა და secret-ის MD5-ის SHA-256. ის ადასტურებს, რომ მოთხოვნა თქვენგან მოდის. მოთხოვნა არასწორი hash-ით, ან id-ით, რომელიც აქტიური ინტეგრაცია არ არის, HTTP 401 პასუხს იღებს.

მაგალითი

```
// 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. ანგარიშის შემოწმება: GET /account/get

ამოწმებს, არის თუ არა ელ. ფოსტა უკვე დარეგისტრირებული MULTIGURA-ში.

პარამეტრი	ტიპი	აღწერა
`id`	int	ინტეგრაციის იდენტიფიკატორი.
`email`	string	მომხმარებლის ელ. ფოსტა.
`hash`	string	SHA-256 id-დან, email-იდან და secret-ის MD5-იდან.

თუ ელ. ფოსტა თავისუფალია, პასუხში status არის 0 და data ცარიელია. თუ დარეგისტრირებულია, status არის 1: data ანგარიშს მხოლოდ მაშინ შეიცავს, როცა მომხმარებელი თქვენი ინტეგრაციით დარეგისტრირდა, სხვა შემთხვევაში ცარიელია.

ველები company_id და company_name data-ში მაშინ ჩანს, როცა შევსებულია. რეგისტრაციის თარიღი (registered) Asia/Tbilisi-ის სასაათო სარტყელშია (UTC+4).

მაგალითი

```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. ანგარიშის შექმნა: POST /account/set

ქმნის ანგარიშს უფასო ერთწლიანი ლიცენზიით. მოთხოვნის ტანი შეიძლება გაიგზავნოს ფორმის მონაცემებად ან JSON-ად.

პარამეტრი	ტიპი	აღწერა
`id`	int	ინტეგრაციის იდენტიფიკატორი.
`user`	string	სასურველი login. ის პატარა ასოებად გადაიყვანება და მასში მხოლოდ ლათინური ასოები და ციფრები რჩება. თუ login ციფრით იწყება, თავში user ემატება, ხოლო თუ დაკავებულია, ბოლოში ნომერი ემატება, ამიტომ პასუხში login შეიძლება განსხვავდებოდეს.
`email`	string	მომხმარებლის ელ. ფოსტა.
`first_name`	string	არასავალდებულო. სახელი.
`last_name`	string	არასავალდებულო. გვარი.
`company_id`	string	არასავალდებულო. კომპანიის იდენტიფიკატორი.
`company_name`	string	არასავალდებულო. კომპანიის სახელი.
`lang`	string	არასავალდებულო. ანგარიშისა და მისასალმებელი წერილის ენა: en, ge ან ru. ნაგულისხმევად en.
`hash`	string	SHA-256 id-დან, user-იდან, email-იდან და secret-ის MD5-იდან. მნიშვნელობა user გამოიყენეთ ზუსტად ისე, როგორც აგზავნით.

თუ ელ. ფოსტა თავისუფალია, MULTIGURA ქმნის ანგარიშს და მის ლიცენზიას, ხოლო თუ თქვენს ინტეგრაციას პრომო კოდი აქვს, კოდს ანგარიშს მიაბამს. მომხმარებელი არჩეულ ენაზე იღებს წერილს პაროლის დასაყენებელი ბმულით. პასუხში status არის 0, data კი ახალ ანგარიშს შეიცავს.

თუ ელ. ფოსტა უკვე დარეგისტრირებულია, არაფერი იქმნება: პასუხი იგივეა, რაც ანგარიშის შემოწმებისას.

მაგალითი

```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. სტატუსები და შეცდომები

HTTP 200 პასუხს ყოველთვის აქვს status:

სტატუსი	აღწერა	მნიშვნელობა
0	Done	ანგარიშის შემოწმება: ელ. ფოსტა თავისუფალია. ანგარიშის შექმნა: ანგარიში შეიქმნა.
1	The email already exists	ელ. ფოსტა დარეგისტრირებულია. არაფერი იქმნება.
99	Unknown system error	მოთხოვნა ვერ შესრულდა. სცადეთ მოგვიანებით და თუ განმეორდა, დაგვიკავშირდით.

მოთხოვნა, რომლის დამუშავებაც შეუძლებელია, ამის ნაცვლად HTTP შეცდომას იღებს, WordPress-ის სტანდარტულ ფორმატში:

HTTP	კოდი	მიზეზი
400	rest_missing_callback_param	აკლია სავალდებულო პარამეტრი.
400	rest_invalid_param	პარამეტრი არასწორია, მაგალითად ელ. ფოსტა.
401	rest_forbidden	არასწორი hash, ან id არ არის აქტიური ინტეგრაცია.

მაგალითი

```
// 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
    }
}
```