

Canaries API

Endpoints

- v1 – API version 1

Routes

- [Auth](#)
 - [Login](#)
 - [Logout](#)
 - [Get all users](#)
 - [Get a particular user](#)
 - [Add new users](#)
 - [Update a particular user](#)
 - [Delete a particular user](#)
- [Domains](#)
 - [Get all domains](#)
 - [Add new domains](#)
 - [Get domain by uuid](#)
 - [Delete a particular domain](#)
- [Monitored sites](#)
 - [Get all sites](#)
 - [Add new sites](#)
 - [Get monitored site by uuid](#)
 - [Delete a particular site](#)
- [Canary nodes](#)
 - [Get all canary nodes](#)
 - [Generate new fake data for canary node](#)
 - [Re-generate fake data for canary node](#)
 - [Delete fake data for canary node](#)
 - [Create new canary nodes](#)
 - [Update canary node](#)
 - [Delete a particular canary node](#)
- [Mail](#)
 - [Get all mail for a particular canary node](#)

Auth

- [Login](#)
- [Logout](#)
- [Get all users](#)
- [Get a particular user](#)
- [Add new users](#)
- [Update a particular user](#)
- [Delete a particular user](#)

/[endpoint]/auth/login : POST

- description: Log in and get access token
- request:
 - parameters:
 - body:

```
{  
  "username": "user",  
  "password": "secret123"  
}
```

- response:
 - http_code: 200
 - parameters:
 - body:

```
{  
  "token": "JWT_ACCESSTOKEN",  
  "token_type": "bearer",  
  "expires": 3600,  
  "uuid": "uuidstring",  
  "permissions": ["admin"],  
  "canaries": []  
}
```

- response:
 - http_code: 400
 - parameters:
 - http_headers:
 - X-RateLimit-Limit: 10
 - X-RateLimit-Remaining: 7
 - body:

```
{  
  "code": 0,  
  "message": "Bad request",  
  "details": "Invalid credentials..."  
}
```

- response:
 - http_code: 429
 - parameters:
 - http_headers:
 - X-RateLimit-Limit: 10
 - X-RateLimit-Remaining: 0
 - X-RateLimit-Reset: 1582480798

- Retry-After: 60
- body:

```
{
  "code": 1,
  "message": "Rate limit exceeded",
  "retry": 60
}
```

{endpoint}/auth/logout : GET

- decription: Log user out
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
- response:
 - http_code: 200

{endpoint}/auth/users : GET

- decription: Get all users
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - request (with limit - optional): limit=10
 - request (with offset - optional): offset=100
 - request (with limit & offset): limit=5&offset=50
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "count": 10,
  "total": 2314,
  "offset": 100,
  "users": [
    {
      "uuid": "uuidstring1",
      "username": "jozkomrkvicka",
      "permissions": ["admin", "worker"],
      "canaries": [],

```

```
        "created_at": "2020-02-19 08:46:28",
        "updated_at": "2020-02-19 08:46:28",
        "updated_by": null
      },
      {
        "uuid": "uuidstring2",
        "username": "peter",
        "permissions": ["worker"],
        "canaries": ["uuidstring1", "uuidstring2"],
        "created_at": "2020-02-19 11:33:01",
        "updated_at": "2020-02-19 11:33:01",
        "updated_by": "adminuuidstring"
      }
    ]
  }
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Invalid range"
}
```

- description: Get a particular user information
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - {uuid}: Public user identification string

- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "uuid": "uuidstring",
  "username": "jozkomrkvicka",
  "permissions": ["admin", "worker"],
  "canaries": ["uuidstring1", "uuidstring2"],
  "created_at": "2020-02-19 08:46:28",
  "updated_at": "2020-02-19 08:46:28"
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "..."}
}
```

{endpoint}/auth/users : POST

- decription: Create a new user
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - permissions: User permissions - can be: admin, worker or expert
 - body:

```
{
  "users": [
    {
      "username": "peter",
      "password": "heslo",
      "permissions": ["worker"],
      "canaries": ["uuidstring1", "uuidstring2"]
    }
  ]
}
```

- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "users": [
    {
      "username": "peter",
      "permissions": ["worker"],
      "canaries": ["uuidstring1", "uuidstring2"],
      "uuid": "uuidstring",
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28",
      "updated_by": "adminuuidstring"
    }
  ]
}
```

- response:

- http_code: 400
- parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "..."
}
```

`{endpoint}/auth/users/{uuid}`: PUT

- decription: Update a particular user
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESS_TOKEN"
 - **`{uuid}`**: Public user identification string
 - permissions: User permissions – can be: `admin`, `worker` or `expert`
 - body:

```
{
  "username": "peter1",
  "password": "noveheslo",
  "permissions": ["admin"],
  "canaries": ["uuidstring"]
}
```

- response:
 - http_code: 200
- response:
 - http_code: 400
 - parameters:

- body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
- body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
- body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "..."
}
```

`/endpoint/auth/users/{uuid}`: DELETE

- description: Delete a particular user
- request
 - parameters:

- http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - {uuid}: Public user identification string
- response:
 - http_code: 200
- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "...
}
```

{endpoint}/auth/refresh_token : GET

- decription: Get new access token
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "token": "JWT_ACCESSTOKEN",
  "token_type": "bearer",
  "expires": 3600
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

Domains

- [Get all domains](#)
- [Add new domains](#)
- [Get domain by uuid](#)
- [Delete a particular domain](#)

/endpoint/domains : GET

- description: Prints all available canary domains
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - request (with limit - optional): limit=10
 - request (with offset - optional): offset=100
 - request (with limit & offset): limit=5&offset=50
- response:
 - http_code: 200
 - parameters:
 - body:

```
{  
  "count": 10,  
}
```

```
"total": 455,  
"offset": 100,  
"domains": [  
  {  
    "uuid": "uuidstring1",  
    "domain": "domainname.tld",  
    "created_at": "2020-02-19 08:46:28",  
    "updated_at": "2020-02-19 08:46:28"  
  },  
  {  
    "uuid": "uuidstring2",  
    "domain": "another.tld",  
    "created_at": "2020-02-19 08:46:28",  
    "updated_at": "2020-02-19 08:46:28"  
  }  
]  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 2,  
  "message": "Invalid range"  
}
```

/endpoint/domains : POST

- description: Add new domain(s)
- request:
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - body:

```
{  
  "domains": ["domena.sk"]  
}
```

- response:
 - http_code: 200
 - parameters:
 - body:

```
{  
  "domains": [  
    {  
      "uuid": "uuidstring",  
      "domain": "domena.sk",  
      "created_at": "2020-02-19 08:46:28",  
      "updated_at": "2020-02-19 08:46:28"  
    }  
  ]  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."}
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "Domains already exist..."}
}
```

/endpoint/domains/{uuid}: GET

- description: Get domain name by uuid
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - {uuid}: Canary domain uuid
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "uuidstring": "domena.sk"}
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"}
}
```

- response:

- http_code: 401
- parameters:
 - body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 2,  
  "message": "Bad request",  
  "details": "Domain does not exist..."  
}
```

{endpoint}/domains/{uuid}: DELETE

- description: Delete a domain
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - {uuid}: Canary domain uuid

- response:
 - http_code: 200

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:

- body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."}

```

Monitored sites

- [Get all sites](#)
- [Add new sites](#)
- [Get monitored site by uuid](#)
- [Delete a particular site](#)

/[endpoint]/sites : GET

- description: Prints all monitored sites
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - request (with limit - optional): limit=10
 - request (with offset - optional): offset=100
 - request (with limit & offset): limit=5&offset=50
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "count": 10,
  "total": 234,
  "offset": 100,
  "sites": [
    {
      "uuid": "uuidstring1",
      "site": "facebook.com",
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28"
    },
    {
      "uuid": "uuidstring2",
      "site": "azet.sk",
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28"
    }
  ]
}
```

```
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 2,  
  "message": "Invalid range"  
}
```

{endpoint}/sites : POST

- description: Add new site(s)
- request:
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - body:

```
{  
  "sites": ["bazos.sk"]  
}
```


- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "sites": [
    {
      "uuid": "uuidstring",
      "site": "bazos.sk",
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28"
    }
  ]
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "Sites already exist..."
}
```

`/endpoint/sites/{uuid} : GET`

- decription: Get site name by uuid
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - **{uuid}**: Monitored site uuid
- response:
 - http_code: 200
 - parameters:

- body:

```
{  
  "uuidstring": "facebook.com"  
}
```

- response:
 - http_code: 400
 - parameters:

- body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:

- body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

- response:
 - http_code: 400
 - parameters:

- body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "Site does not exist..."
}
```

`/endpoint/sites/{uuid}`: DELETE

- decription: Delete a site
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - **{uuid}**: Monitored site uuid
- response:
 - http_code: 200
- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

Canary nodes

- [Get all canary nodes or canary details](#)
- [Generate new fake data for canary node](#)
- [Re-generate fake data for canary node](#)
- [Delete fake data for canary node](#)
- [Create new canary nodes](#)

- [Update canary node](#)
- [Delete a particular canary node](#)

/endpoint/canaries : GET

- description: Prints all canary nodes available to the user. Account with `expert` permission will get all canaries that have **testing** parameter set to true, and account with `admin` permission will get all canaries. If uuid or email parameters are specified, then only detail of that canary will be returned.
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - request (with limit - optional): limit=10
 - request (with offset - optional): offset=100
 - request (with limit & offset): limit=5&offset=50
 - request (with uuid): uuid=uuidstring
 - request (with email): email=milan.paradajka@domena.tld
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "count": 10,
  "total": 11456,
  "offset": 100,
  "canaries": [
    {
      "uuid": "uuidstring",
      "domain": "uuidstring",
      "site": "uuidstring",
      "assignee": "uuidstring",
      "testing": false,
      "setup": false,
      "email": "milan.paradajka@domena.tld",
      "password": "hesielko123",
      "data": {
        "username": "milan.paradajka",
        "firstname": "Milan",
        "lastname": "Paradajka",
        "phone": "+412 123 456 789"
      },
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28",
      "updated_by": "adminuuidstring"
    }
  ]
}
```

```
}
```

- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "uuid": "uuidstring",
  "domain": "uuidstring",
  "site": "uuidstring",
  "assignee": "uuidstring",
  "testing": false,
  "setup": true,
  "email": "milan.paradajka@domena.tld",
  "password": "hesielko123",
  "data": {
    "username": "milan.paradajka",
    "firstname": "Milan",
    "lastname": "Paradajka",
    "phone": "+412 123 456 789"
  },
  "created_at": "2020-02-19 08:46:28",
  "updated_at": "2020-02-19 08:46:28",
  "updated_by": "adminuuidstring"
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:

- http_code: 400
- parameters:
 - body:

```
{
  "code": 2,
  "message": "Invalid range"
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 3,
  "message": "Bad request",
  "details": "Canary does not exist..."
}
```

`/endpoint/canaries/{uuid}/{parameter}`: GET

- description: Generates fake information for the canary node registration process
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - **`{uuid}`**: Canary node uuid
 - **`{parameter}`**: Parameter name
 - username
 - firstname
 - lastname
 - birthday
 - sex
 - address
 - phone
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "parameter": "value"
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "..."
}
```

`/endpoint/canaries/{uuid}/{parameter}`: POST

- description: Re-generates fake information
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESS_TOKEN"
 - **`{uuid}`**: Canary node uuid
 - **`{parameter}`**: Parameter name
 - username
 - firstname
 - lastname
 - birthday
 - sex

- address
 - phone
- response:
 - http_code: 200
 - parameters:
 - body:

```
{  
  "parameter": "value"  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{  
  "code": 2,  
  "message": "Bad request",  
  "details": "..."  
}
```

`/endpoint/canaries/{uuid}/{parameter}`: DELETE

- description: Deletes fake information

- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - {uuid}: Canary node uuid
 - {parameter}: Parameter name
 - username
 - firstname
 - lastname
 - birthday
 - sex
 - address
 - phone
- response:
 - http_code: 200
- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

{endpoint}/canaries : POST

- description: Create new canary nodes
- request:
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - password strength (optional - default random):

- `dictionary` - random password from the top 1 million
- `simple` - word + word(p=0.7) + digit + digit(p=0.5) + digit(p=0.1)
- `random` - 8 random characters from [a-zA-Z0-9]
- `strong` - 13 - 18 totally random characters
- `trivial` - word + digit + digit(p=0.5)
- body:

```
{
  "domain": "uuidstring",
  "site": "uuidstring",
  "testing": false,
  "count": 10,
  "password_strength": "simple"
}
```

- response:
 - `http_code`: 200
 - parameters:
 - body:

```
{
  "canaries": [
    {
      "uuid": "uuidstring",
      "domain": "uuidstring",
      "site": "uuidstring",
      "assignee": "uuidstring",
      "testing": false,
      "setup": false,
      "email": "milan.paradajka@domena.tld",
      "password": "hesielko123",
      "data": {
        "username": "milan.paradajka"
      },
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28",
      "updated_by": "adminuuidstring"
    }
  ]
}
```

- response:
 - `http_code`: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."/>

```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Bad request",
  "details": "..."/>

```

`/endpoint/canaries/{uuid}`: PUT

- description: Update canary information. Changing of every, but setup, parameter needs admin permission. Setup parameter can be updated by the assignee.
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - **{uuid}**: Canary node uuid
 - site
 - assignee
 - testing
 - setup
- response:
 - http_code: 200
- response:
 - http_code: 400

- parameters:

- body:

```
{  
  "code": 0,  
  "message": "Token not provided"  
}
```

- response:

- http_code: 401

- parameters:

- body:

```
{  
  "code": 1,  
  "message": "Unauthorized",  
  "details": "..."  
}
```

- response:

- http_code: 400

- parameters:

- body:

```
{  
  "code": 2,  
  "message": "Bad request",  
  "details": "..."  
}
```

`/endpoint/canaries/{uuid}`: DELETE

- description: Delete a canary node

- request

- parameters:

- http_headers:

- Authentication: "bearer JWT_ACCESSTOKEN"

- **`{uuid}`**: Canary node uuid

- response:

- http_code: 200

- response:

- http_code: 400

- parameters:

- body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "...
}
```

Mail

- [Get all mail for a particular canary node](#)

`/endpoint/mail/{uuid}` : GET

- description: Prints all e-mails received by a particular canary account
- request
 - parameters:
 - http_headers:
 - Authentication: "bearer JWT_ACCESSTOKEN"
 - {uuid}: Canary node uuid
- response:
 - http_code: 200
 - parameters:
 - body:

```
{
  "emails": [
    {
      "uuid": "uuidstring",
      "canary": "uuidstring",
      "received_on": "2020-02-19 08:45:00",
      "from": "sender@domain.tld",
      "subject": "message subject",
      "body": "raw body",
      "created_at": "2020-02-19 08:46:28",
      "updated_at": "2020-02-19 08:46:28"
    }
  ]
}
```

-
- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 0,
  "message": "Token not provided"
}
```

- response:
 - http_code: 401
 - parameters:
 - body:

```
{
  "code": 1,
  "message": "Unauthorized",
  "details": "..."
}
```

- response:
 - http_code: 400
 - parameters:
 - body:

```
{
  "code": 2,
  "message": "Invalid range"
}
```