

Руководство пользователя

Documentation API

Authenticating requests

Authenticate requests to this API's endpoints by sending a X-Auth-Keyb> header with the value "{YOUR_AUTH_KEY}".

All authenticated endpoints are marked with a requires authentication badge in the documentation below.

You can get a token along the route /api/login (by nick and password) or /api/token (by exist token) with POST method

Tokens

Token generation

Requires authentication.

Request: POST api/token.

Body Parameters:

1. name string — Token name.

```
curl --request POST \
  "http://localhost/api/token" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"name\": \"report-service\"
  }"
```

Example request

```
{
```

```

"data": {
  "name": "report-service",
  "token": "{YOUR_AUTH_KEY}"
}
}

```

Example response (201)

Read all authorized user tokens

Requires authentication.

Request: GET api/token .

```

curl --request GET \
  --get "http://localhost/api/token" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json"

```

Example request

```

{
  "data": {
    "tokens": [
      {
        "name": "login",
        "abilities": [
          "*"
        ],
        "last_used_at": "2022-04-01T11:17:50.000000Z"
      },
      {
        "name": "report-service",
        "abilities": [
          "*"

```

```

    ],
    "last_used_at": null
  }
]
}
}

```

Example response (200)

Read an authorized user tokens

Requires authentication.

Request: GET api/token/{token_name}.

URL Parameters:

1. token_name string — user token name

```

curl --request GET \
  --get "http://localhost/api/token/report-service" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json"

```

Example request

```

{
  "data": {
    "name": "report-service",
    "abilities": [
      "*"
    ],
    "last_used_at": null
  }
}

```

Example response (200)

Edit token

Requires authentication.

Edit token name and rights. Rights will be used only those that are available to the user of the token.

Token key stays the same.

Request: PUT api/token/{token_name}.

URL Parameters:

1. token_name string — user token name

Body Parameters:

1. name string — token name
2. abilities string[] optional — list of rules

```
curl --request PUT \
  "http://localhost/api/token/report-service" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"name\": \"report-service\",
    \"abilities\": [
      \"*\
    ]
  }"
```

Example request

```
{
  "data": {
    "name": "report-service",
    "abilities": [
      "report:getShort",
```

```

        "report:getByProject"
    ],
    "last_used_at": null
}
}

```

Example response (200)

Deleting a token

Requires authentication.

Removing an authorized user token by token name.

Request: DELETE api/token/{token_name}.

URL Parameters:

1. token_name string — user token name

```

curl --request DELETE \
    "http://localhost/api/token/report-service" \
    --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
    --header "Content-Type: application/json" \
    --header "Accept: application/json"

```

Example request

[Empty response]

Example response (204)

Users

Authentication

User authentication. A token is issued for the user's nick and password for further interaction with the api.

The token is created under the name "login" and is available like other tokens.

If a "login" token previously existed, it is removed and a new user token is generated.

Request: POST api/login.

Body Parameters:

1. nick string — user nick
2. password string — user password

```
curl --request POST \
  "http://localhost/api/login" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"nick\": \"testuser\",
    \"password\": \"password\"
  }"
```

Example request

```
{
  "data": {
    "name": "login",
    "token": "{YOUR_AUTH_KEY}"
  }
}
```

Example response (200)

Report hours

Short report

Requires authentication.

Getting a short report on hours from employees for a certain period of days.

Request: GET api/report/hours/short.

Body Parameters:

1. `start_date` string optional — Start date of the reporting period. Inclusive. Must be a valid date.

2. `end_date` string optional — End date of the reporting period. Inclusive. Must be a valid date.

```
curl --request GET \
  --get "http://localhost/api/report/hours/short" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"start_date\": \"2010-10-07\",
    \"end_date\": \"2010-10-09\"
  }"
```

Example request

```
{
  "data": {
    "count_reports": 1,
    "reports": [
      {
        "nick": "ilene.wolf",
        "name": "Vickie Jerde",
        "company": "P",
        "total_hours": 5
      }
    ]
  }
}
```

Example response (200)

Report hours by project

Requires authentication.

Getting a report on hours from employees ordered by project for a certain period of days.

If several employees worked on one project, then several records are created in the reports array with the same value of the fields related to the project.

Request: GET api/report/hours/project.

Body Parameters:

1. start_date string optional — Start date of the reporting period. Inclusive. Must be a valid date.

2. end_date string optional — End date of the reporting period. Inclusive. Must be a valid date.

```
curl --request GET \
  --get "http://localhost/api/report/hours/project" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"start_date\": \"2010-10-07\",
    \"end_date\": \"2010-10-09\"
  }"
```

Example request

```
{
  "data": {
    "count_reports": 2,
    "reports": [
      {
        "nick": "ilene.wolf",
```



```

    "name": "Vickie Jerde",
    "company": "P",
    "customer_id": 1,
    "customer_name": "ullam et esse",
    "project_id": 1,
    "project_name": "pariatur voluptatem corrupti",
    "total_hours": 3.2
  },
  {
    "nick": "ilene.wolf",
    "name": "Vickie Jerde",
    "company": "P",
    "customer_id": 2,
    "customer_name": "dolorem aut nostrum",
    "project_id": 2,
    "project_name": "voluptas quas labore",
    "total_hours": 1.8
  }
]
}
}

```

Example response (200)

Schedule

Official schedule

Requires authentication.

Getting a short report on hours from employees for a certain period of days.

Request: GET api/schedule/official.

Body Parameters:

1. `start_date` string optional — Start date of the reporting period. Inclusive. Must be a valid date.

2. `end_date` string optional — End date of the reporting period. Inclusive. Must be a valid date.

```
curl --request GET \
  --get "http://localhost/api/shedule/official" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"start_date\": \"2010-10-07\",
    \"end_date\": \"2010-10-09\"
  }"
```

Example request

```
{
  "data": {
    "2010-10-07": {
      "status": "holiday",
      "info": [
        "Error possimus maiores eius est sit molestiae qui qui."
      ]
    },
    "2010-10-08": {
      "status": "holiday",
      "info": [
        "Error possimus maiores eius est sit molestiae qui qui."
      ]
    },
    "2010-10-09": {
      "status": "work",
```

```

        "info": []
    }
}

```

Example response (200)

Workers schedule

Requires authentication.

Getting the workers schedule for the period requested with his worked hours by days.

Request: GET api/shedule/worker.

Body Parameters:

1. start_date string optional — Start date of the reporting period. Inclusive. Must be a valid date.

2. end_date string optional — End date of the reporting period. Inclusive. Must be a valid date.

```

curl --request GET \
  --get "http://localhost/api/shedule/worker" \
  --header "X-Auth-Key: {YOUR_AUTH_KEY}" \
  --header "Content-Type: application/json" \
  --header "Accept: application/json" \
  --data "{
    \"start_date\": \"2010-10-07\",
    \"end_date\": \"2010-10-09\"
  }"

```

Example request

```

{
  "ilene.wolf": {
    "2010-10-07": {

```

```

    "status": "holiday",
    "info": [
        "Error possimus maiores eius est sit molestiae qui qui."
    ]
},
"2010-10-08": {
    "status": "work",
    "info": [
        "Error possimus maiores eius est sit molestiae qui qui.",
        "Provident delectus et atque sequi."
    ],
    "total_hours": 5
},
"2010-10-09": {
    "status": "work",
    "info": []
}
}
}

```

Example response (200)