HAKATON

API v1 for consumption by Hackaton Company SA | Electronic Point
Requirements
To use our API, you must be familiar with the basic terminology used in developing REST web services. It is possible to develop integration with practically all programming languages available on the market.
Tests
You can test the API using forms in the interactive documentation itself.
We also recommend using Postman, a free Chrome plugin that makes it easy to inspect requests and responses.
Format
The API is a web service compatible with REST, that is, in addition to the traditional POST and GET methods, you will also see operations here that use the PUT and DELETE methods. Requests are returned in JSON format only, always accompanied by the appropriate HTTP code.
Security and authentication
Authentication to the API is carried out using a valid Bearer Token generated through a request to send a username and password*.

Overview

To use the API, you must make Rest API calls to use the services. To use the API, the user must make the authentication request (User/Login) and send the generated token to bearerToken. such as timekeeping and reports.

Rate Limits

The API follows commercial use case rate limits. Each customer account has a call count rate limit and each call made by its channel instance.

User

The APIs aimed at the user group offer functionality for creating new users in the time recording system, in addition to allowing these users to log in.

POST Signup

/dev/api/v1/user/signup

Returns the result of creating a user.

Request Parameters

Name	Type	Description
email	string (255)	email user
password	string (24)	

Response Parameters

the return is an array of objects with the response parameters below

Name	Type	Description
message	string (2024)	

Body raw (json)

```json
{
    "email": "eliza@andersonalves.com.br",
    "password": "Teste@123456",
    "name": "Eliza M Alves",
    "matricula": "123456"
}
```

POST Login

/dev/api/v1/user/login

Returns the access_token and refresh_token, the access_token must be sent in bearerToken in other Report and Point Report requests.

Request Parameters

Name	Type	Description
email	string (255)	email user
password	string (24)	

Response Parameters

the return is an array of objects with the response parameters below

Name	Type	Description
access_token	string (2024)	
refresh_token	string (2024)	

Body raw (json)

```
json

{
    "email": "anderson@andersonalves.com.br",
    "password": "Teste@123456"
}
```

Report

The APIs aimed at the reporting group enable the request to generate the point mirror and allow querying the status of the order.

POST RequestReport

/dev/api/v1/report/request

Makes the request to generate the point mirror forwarding via email.

Request Parameters

Name	Type	Description
type	string(60)	fixed espelho
competence	string (7)	sample 2024-03

Response Parameters

returns an object

Name	Type	Description
id	string (36)	

AUTHORIZATION Bearer Token

AUTHORIZATION Bearer Token

Token	{{TOKEN}}

Body raw (json)

```json
{
    "type": "espelho",
    "competence": "2024-03"
}
```

GET GetRequestReport

/dev/api/v1/report/e62f4232-8ad2-47b3-9ca3-adf1720df1ce

Returns how a point mirror report request is being processed.

Path Parameters

Name	**Type**	Description
id	string(36)	

Response Parameters

returns an array of objects

Name	**Type**	Description
report	array	

Name	**Type**	Description
userId	string(36)	
competence	string(7)	
status	string(30)	
createdAt	datetime	
id	string(36)	
type	string(60)	

AUTHORIZATION Bearer Token

Token	{{TOKEN}}

Point Record

The APIs related to the timekeeping group make it possible to make timekeeping appointments and also allow you to view the history of these timekeeping appointments for a specific period.

GET RecordView

/dev/api/v1/point/view?startDate=2024-03-01&endDate=2024-03-19

Returns the mirror of appointments by period.

Query Parameters

Name	**Type**	Description
startDate	string(10)	
endDate	string(10)	

Response Parameters

returns an array of objects

Name	**Type**	Description
point	object	

Name	**Type**	Description
#date	string(10)	

Name	**Type**	Description
duration	int	
durationBreak	int	
marking	array	

Name	**Type**	Description
type	string(40)	
marking	datetime	
action	string(40)	

AUTHORIZATION Bearer Token

Token	{{TOKEN}}

PARAMS

startDate	2024-03-01
endDate	2024-03-19

POST RegisterPoint

/dev/api/v1/point/entry

Performs user timekeeping.

Response Parameters

returns an array of objects

Name	**Type**	Description
point	object	

Name	**Type**	Description
Attributes	object	

Name	**Type**	Description
date	string(10)	
points	array	

Name	**Type**	Description
dateHour	datetime	

AUTHORIZATION Bearer Token

Token	{{TOKEN}}