Logisteps Rest Framework v1

# Overview

This document describes high level documentation for accessing and manipulating data for the Logisteps application. All permanent data is stored and accessed via the Logisteps backend webserver, which makes reads and writes to the Logisteps database. Database schema details can be found in other documentation.

# Current Version

This the first version of the Logisteps rest API. Updates to the rest API, including modifications, additions, and deletions will be documented in subsequent versions. The latest version of this documents can be determined by accessing the following file:

<https://github.com/SeniorDesignTeamOmicron/Documentation/blob/master/Project%20Design/web%20application/current/current%20version.txt>

# Schema

All access to the rest API will be performed using the HTTPS protocol, at <https://logisteps.com/api>. All data sent to the server and data returned from the server will sent using the JSON format.

curl -X GET \

> http://127.0.0.1:8000/api/user/larsonma/ \

> -H 'Authorization: Basic bGFyc29ubWE6dGVzdA==' \

> -H 'Postman-Token: adb8f77e-4b8c-4bd4-ba18-ab59eec1839b' \

> -H 'cache-control: no-cache'

HTTP/1.1 200 OK

Date: Sat, 12 Jan 2019 22:26:24 GMT

Server: WSGIServer/0.2 CPython/3.7.0

Content-Type: application/json

Vary: Accept, Cookie

Allow: GET, PUT, DELETE, HEAD, OPTIONS

X-Frame-Options: SAMEORIGIN

Content-Length: 229

All datetime objects are to be sent and received using the ISO-8601 format using UTC time with offsets for timezone differences:

2019-01-12T19:02:19+00:00

# Authentication

Several API endpoints require authentication to both view and manipulate data. In the initial design of the API, only one form of authentication is supported: basic authentication. This will require clients to temporarily store a username password pair and pass them to the API in each request.

## Basic Authentication

In order to access protected endpoints, basic authentication must be provided in the header of the request, encoded using base 64.

curl -X GET \ http://127.0.0.1:8000/api/user/larsonma/ \

> -H Authorization: Basic bGFyc29ubWE6dGVzdA==

The authorization data is decoded on the server and used to validate against current user profiles.

## Failed Authentication

If a client attempts to access a resource which required authentication, and the client provides incorrect credentials, a 403 Forbidden response from the server will be returned with a JSON object indicating that the username or password was incorrect.

curl -i -X GET \

> http://127.0.0.1:8000/api/user/larsonma/ \

> -H 'Authorization: Basic bGFyc29ubWE6dGVzdHQ='

HTTP/1.1 403 Forbidden

Date: Sat, 12 Jan 2019 22:49:19 GMT

Server: WSGIServer/0.2 CPython/3.7.0

Content-Type: application/json

Vary: Accept, Cookie

Allow: GET, PUT, DELETE, HEAD, OPTIONS

X-Frame-Options: SAMEORIGIN

Content-Length: 39

{"detail":"Invalid username/password."}

## Failed Login Limit

There is currently no failed login limit, but this may be implemented in a future version.

# Parameters

Some requests to the rest API allow for optional parameters. When sending GET requests, the parameters should be appended to the end of the URL following a ‘?’. Each parameter should be a key value pair.

curl -i http://127.0.0.1:8000/api/steps/steplist/?date=07-04-2018

In this GET request, a date parameter of 07-04-2018 is send by appending to the end of the URL.

For POST, PUT, and DELETE requests, parameters should be passed using a JSON object in the request’s body.

# Client Errors

Processing errors that are the result of a client mistake will always send a 400 Bad Request response. The body of the response will contain details regarding the source of the response.

HTTP/1.1 400 Bad Request

Content-Length: 55

[{"location":{"latitude":["This field is required."]}}]

# Users

This is a readable and writeable API for Logisteps users. This API can be leveraged to create, update, get, and delete user objects. This API provides clients the ability to register new users, authenticate users, and retrieve/update details.

## Create a User

Any client may register a new user using this endpoint, as it requires no authentication. Creating a user creates a new LogistepsUser instance in the backend database.

New users must have a unique username not present in the database. Attempting to create a user using a username that already exists will result in a 400 Bad Request with a JSON body describing the issue.

POST /api/user/

### Authorization

Not Required

### Parameters

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| user | Object | JSON object containing user details |
| user.username | String | Unique username for identifying user |
| user.email | String | Valid email that user can be contacted at |
| user.first\_name | String | First name of the user |
| user.last\_name | String | Last name of the user |
| user.password | String | Plain text password for user. Will be encrypted prior to being stored |
| right\_shoe | Object | JSON object containing details for the user’s right shoe |
| right\_shoe.foot | Char | Character representing which foot the shoe belongs to. Valid options are “R” or “L” |
| right\_shoe.size | Float | Size of the user’s right foot. Precision should be limited to 1 decimal. Valid sizes are between 4 and 16. |
| left\_shoe | Object | JSON object containing details for the user’s left shoe |
| left\_shoe.foot | Char | Character representing which foot the shoe belongs to. Valid options are “R” or “L” |
| left\_shoe.size | Float | Size of the user’s left foot. Precision should be limited to 1 decimal. Valid sizes are between 4 and 16. |
| height | Integer | Height of the user in inches |
| Weight | Integer | Weight of the user in pounds |
| step\_goal | Integer | User’s daily step goal. Must be greater than or equal to 0. |

### Response

Status: 201 Created

{

"user": {

"id": 5,

"username": "larsonm22",

"email": "test@aol.com",

"first\_name": "Mitchell",

"last\_name": "Larson"

},

"left\_shoe": {

"size": "8.5",

"foot": "L"

},

"right\_shoe": {

"size": "8.5",

"foot": "R"

},

"height": 67,

"weight": 165,

"step\_goal": 10000

}

## Get a User

This endpoint allows a client to retrieve a user object. This may be useful for loading user profile details or verifying user credentials at login. If valid credentials are provided, the entire user object, except the password, is returned in the response.

GET /api/user/<username>/

### Authorization

Required

### Parameters

None

### Response

Status: 200 OK

{

"user": {

"id": 5,

"username": "larsonm22",

"email": "test@aol.com",

"first\_name": "Mitchell",

"last\_name": "Larson"

},

"left\_shoe": {

"size": "8.5",

"foot": "L"

},

"right\_shoe": {

"size": "8.5",

"foot": "R"

},

"height": 67,

"weight": 165,

"step\_goal": 10000

}

## Get List of All Users

Allows a client to obtain a list of all user profiles in the Logisteps database. This endpoint is intended for administrative purposes and requires special privileges for access.

GET /api/users/

### Authorization

Requires admin privileges. The client must be a user registered with is\_staff set to true. It is not currently possible to create a user with such privileges using the API. Users with this privilege must be created using the command line interface.

### Parameters

None

### Response

Status: 200 OK

[{

"user": {

"id": 3,

"username": "larsonma",

"email": "larsonma@aol.com",

"first\_name": "Mitchell",

"last\_name": "Larson"

},

"left\_shoe": {

"size": "9.5",

"foot": "L"

},

"right\_shoe": {

"size": "9.5",

"foot": "R"

},

"height": 67,

"weight": 185,

"step\_goal": 49

},

…,

]

## Update a User

Allows a client to update a user’s profile. This is useful in the case that a user may want to update their step goal, change their weight, or even change shoe details.

Note: to update a user, the entire user object must be sent to the server. PATCH is not currently supported.

PUT /api/user/<username>/

### Authentication

Required

### Parameters

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| user | Object | JSON object containing user details |
| user.email | String | Valid email that user can be contacted at |
| user.first\_name | String | First name of the user |
| user.last\_name | String | Last name of the user |
| user.password | String | Plain text password for user. Will be encrypted prior to being stored |
| right\_shoe | Object | JSON object containing details for the user’s right shoe |
| right\_shoe.foot | Char | Character representing which foot the shoe belongs to. Valid options are “R” or “L” |
| right\_shoe.size | Float | Size of the user’s right foot. Precision should be limited to 1 decimal. Valid sizes are between 4 and 16. |
| left\_shoe | Object | JSON object containing details for the user’s left shoe |
| left\_shoe.foot | Char | Character representing which foot the shoe belongs to. Valid options are “R” or “L” |
| left\_shoe.size | Float | Size of the user’s left foot. Precision should be limited to 1 decimal. Valid sizes are between 4 and 16. |
| height | Integer | Height of the user in inches |
| Weight | Integer | Weight of the user in pounds |
| step\_goal | Integer | User’s daily step goal. Must be greater than or equal to 0. |

### Response

Status: 200 OK

{

"user": {

"id": 3,

"username": "larsonma",

"email": "larsonma@aol.com",

"first\_name": "Mitchell",

"last\_name": "Larson"

},

"left\_shoe": {

"size": "9.5",

"foot": "L"

},

"right\_shoe": {

"size": "9.5",

"foot": "R"

},

"height": 67,

"weight": 185,

"step\_goal": 49

}

## Delete a User

Allows a user to be deleted from the Logisteps database. While this will delete a user’s profile from the application, this will not delete a user’s step data. Their step data will remain in the database with an anonymous user.

DELETE /api/user/<username>/

### Authentication

Required

### Parameters

None

### Response

Status: 204 No Content

# Steps

This is a readable and writeable API for creating and reading step data pertaining to a user. This API also provides a method for obtaining summary statistics regarding a user’s step data for a particular day.

## Post Step Data

Allows a client to post step data for a user. This endpoint is designed to allow multiple instances of step data to be posted in a single request, reducing the amount of HTTP overhead needed to post multiple samples. This endpoint will determine which user to save the step data for based on the authorization header of the request.

POST /api/steps/

### Authorization

Required

### Parameters

Parameters detailed below should be provided as a JSON object

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
|  | Array | Unnamed JSON array of step data instances |
|  | Object | Unnamed JSON representing step data |
| datetime | String | String representation of an ISO-8601 datetime object with timezone offset. |
| sensor\_reading | Object | JSON object containing sensor data |
| sensor\_reading.location | Char | Character representing the location of the sensor on the shoe. “T” for top, “B” for bottom. |
| sensor\_reading.pressure | Float | Pressure recorded for the step |
| sensor\_reading.shoe | String | Indication of which shoe the step was taken. “right” for the right shoe, “left” for the left shoe. |
| right\_shoe.size | Float | Size of the user’s right foot. Precision should be limited to 1 decimal. Valid sizes are between 4 and 16. |
| location | Object | JSON object containing details for the user’s location when the step was taken. |
| location.latitude | Float | Latitude of user’s location |
| location.longitude | Float | Longitude of user’s location |

### Response

Status: 201 Created

[

{

"datetime": "2018-07-01T15:39:47-05:00",

"sensor\_reading": {

"pressure": 17.0183,

"location": "B",

"shoe": "ID: 5, size: 9.5, foot: L"

},

"location": {

"latitude": 178.92323,

"longitude": -9.040912

}

},

…,

]

## Get Step Data for a Day

Allows a client to retrieve step data for a particular day.

GET /api/steps/stepslist/?date=mm-dd-yyyy

### Authorization

Required

### Parameters

An optional date parameter can be appended to the end of the URL to specify which date the client would like to receive step data for. If no parameter is provided, the default behavior will return the step data for the current day.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| [date] | String | The day that should be used to filter steps. Required format is mm-dd-yyyy |

### Response

Status: 200 OK

[

{

"datetime": "2018-07-04T20:15:21-05:00",

"sensor\_reading": {

"pressure": 20.1686,

"location": "B",

"shoe": "ID: 5, size: 9.5, foot: L"

},

"location": {

"latitude": -16.1412,

"longitude": 147.132778

}

},

{

"datetime": "2018-07-04T23:25:32-05:00",

"sensor\_reading": {

"pressure": 10.289,

"location": "T",

"shoe": "ID: 5, size: 9.5, foot: L"

},

"location": {

"latitude": -60.335316,

"longitude": -55.429122

}

},

{

"datetime": "2018-07-04T18:22:49-05:00",

"sensor\_reading": {

"pressure": 18.723,

"location": "B",

"shoe": "ID: 6, size: 9.5, foot: R"

},

"location": {

"latitude": 45.445237,

"longitude": 134.806769

}

}

]

## Get Step Summary

Allows a client of obtain a summary of steps statistics for a given day. Clients can specify the date in which they would like a step summary by appending a date parameter to the end of the URL.

GET /api/steps/summary/?date=mm-dd-yyyy

### Authentication

Required

### Parameters

An optional date parameter can be appended to the end of the URL to specify which date the client would like to receive step data for. If no parameter is provided, the default behavior will return the step data for the current day.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| [date] | String | The day that should be used to filter steps. Required format is mm-dd-yyyy |

### Response

Status: 200 OK

{

"steps": 33,

"goal": 49,

"percent": 67.3469387755102,

"least\_active": {

"hour": 2,

"steps": 1

},

"most\_active": {

"hour": 0,

"steps": 5

},

"inactive\_time": {

"hours": 23,

"minutes": 29

},

"steps\_per\_hour": 1.375

}

## Get Step Count(s)

Allows a client of obtain the total steps taken by a user for a range of time. This will only return the total amount of steps taken by a user for each day in a range of days and will not return the instances of the step data.

GET /api/steps/count/?start=mm-dd-yyyy;end=mm-dd-yyyy

### Authentication

Required

### Parameters

This endpoint accepts two optional parameters. When using this endpoint, either the parameters should both be omitted, or both included; a client should not supply only one of the two optional parameters. The end date should not be in the future. There should also be no more than 365 days between the start and end date. If no parameters are provided, the endpoint will return the step count for the current day.

Additionally, the start date must occur before the end date.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| [start] | String | The start day that should be used to filter steps. Required format is mm-dd-yyyy |
| [end] | String | The end day that should be used to filter steps. Required format is mm-dd-yyyy |

### Response

Status: 200 OK

{

"range": {

"start": "10-22-2018",

"end": "10-26-2018"

},

"counts": {

"10-22-2018": 9435,

"10-23-2018": 2345,

"10-24-2018": 5555,

"10-25-2018": 6483,

"10-26-2018": 5093

}

}

## Get Breakdown

Allows a client to obtain a breakdown of a user’s activity for the current year. The only supported granularity is weekly, but this endpoint is being designed to handle multiple granularities. Currently, this will return step count grouped by the granularity specified and active time vs. inactive time grouped by the granularity.

GET /api/steps/breakdown/?groupby=weekly

### Authorization

Required

### Parameters

An optional groupby parameter should be provided. If this is not specified, it defaults to weekly. This indicated to the server how the data should be grouped by in it’s query.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| [groupby] | String | Groupby parameter. Weekly, Monthly, or Yearly. |

### Response

Status: 200 OK

{

"groupby": "weekly",

"week": [

{

"day": "Sunday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

},

{

"day": "Monday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

},

{

"day": "Tuesday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

},

{

"day": "Wednesday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

},

{

"day": "Thursday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

},

{

"day": "Friday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

},

{

"day": "Saturday",

"steps": 45235,

"active time": 3542,

"inactive time": 3245

}

]

}

## Get Pressure Snapshot

Allows a client to obtain day aimed at describing changes in pressure a user places on certain points of the foot, over the course of the past month, week, and day.

Logisteps only currently supports a single sensor per foot, but this endpoint is designed to account for future expansion of sensor count. Data returned is an aggregate average.

GET /api/steps/pressure/?date=mm-dd-yyyy

### Authorization

Required

### Parameters

The only parameter is an optional date parameter. If not present, the date defaults to the current day.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| [date] | String | Day to use when calculating average over the period of time. For example, a date of 07-22-2018 would return the average pressure for the week of 07-16-2018. Required format is mm-dd-yyyy |

### Response

Status: 200 OK

{

"query\_date": "07-18-2018",

"pressure": {

"past\_day": 15.9432,

"past\_week": 16.4353,

"past\_month": 18.3453

}

}

## Get Location Data

This endpoint allows a client to request a user’s location data for a single day. Clients should expect an array containing longitude, latitude pairs to be returned. This endpoint will currently not return time data with location data to protect the privacy of users, but this is subject to change in future API revisions.

GET /api/steps/location/?date=mm-dd-yyyy

### Authorization

Required

### Parameters

An optional date parameter can be appended to the end of the URL to specify which date the client would like to receive step location data for. If no parameter is provided, the default behavior will return the step location data for the current day.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| [date] | String | The day that should be used to filter steps location. Required format is mm-dd-yyyy |

### Response

Status: 200 OK

{

"query\_date": "07-18-2018",

"locations": [

{

"latitude": 178.92323,

"longitude": -9.040912

},

{

"latitude": 178.92323,

"longitude": -9.040912

},

{

"latitude": 178.92323,

"longitude": -9.040912

},

{

"latitude": 178.92323,

"longitude": -9.040912

}

]

}