

Hat-Tip API

This an API for Hat-Tip
More information: <https://helloreverb.com>
Contact Info: tye.hickman89@gmail.com
Version: 1.0.0
BasePath:/TyeHickman/hattip.api/1.0.0
Apache 2.0
<http://www.apache.org/licenses/LICENSE-2.0.html>

Access

Methods

[[Jump to Models](#)]

Table of Contents

[Default](#)

- [POST /entries](#)
- [POST /journals](#)
- [POST /users](#)
- [GET /journals](#)
- [GET /entries](#)
- [GET /users/{email}](#)

Default

POST /entries

Up

adds an entry to a Journal (**addEntry**)

Adds an entry to the system

Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

Request body

journalEntry [JournalEntry](#) (optional)
Body Parameter — User info to add

Query parameters

journalId (required)
Query Parameter — pass a journal id string for looking up that journal's entries

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

201

entry added

400

invalid input, object invalid

409

somehow, this entry already exists...

POST /journals

[Up](#)

adds a Journal (**addJournal**)

Adds a Journal to the system

Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

Request body

journalInfo [Journal](#) (optional)

Body Parameter — Journal info to add

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

201

user created

400

invalid input, object invalid

409

user already exists

POST /users

[Up](#)

adds a User item (**addUser**)

Adds a User to the system

Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

Request body

userInfo [User](#) (optional)

Body Parameter — User info to add

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

201

user created

400

invalid input, object invalid

409

user already exists

GET /journals

finds a journal (**getJournal**)

By passing in the appropriate options, you can retrieve a journal

Query parameters

journalId (required)

Query Parameter — pass a journal id from a user

Return type

Object

Example data

Content-Type: application/json

```
"{}"
```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200

there was a journal [Object](#)

400

bad input parameter

GET /entries

searches entries (**searchEntries**)

By passing in the appropriate options, you can search for available users in the system

Query parameters

journalId (required)

Query Parameter — pass a journal id string for looking up that journal's entries

skip (optional)

Query Parameter — number of records to skip for pagination format: int32

limit (optional)

Query Parameter — maximum number of records to return format: int32

Return type

array[[JournalEntry](#)]

Example data

Content-Type: application/json

```
[ {
  "entryBody" : "entryBody",
  "dateCreated" : "2021-08-22T01:02:03.001+0000",
  "entryTitle" : "entryTitle",
  "entryId" : "d290f1ee-6c54-4b01-90e6-d701748f0851"
}, {
  "entryBody" : "entryBody",
  "dateCreated" : "2021-08-22T01:02:03.001+0000",
  "entryTitle" : "entryTitle",
  "entryId" : "d290f1ee-6c54-4b01-90e6-d701748f0851"
} ]
```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses**200**

entries matching criteria

400

bad input parameter

GET /users/{email}

[Up](#)

searches users (**searchUsers**)

By passing in the appropriate options, you can search for available users in the system

Path parameters**email (required)**

Path Parameter — pass an email string for looking up users

Query parameters**skip (optional)**

Query Parameter — number of records to skip for pagination format: int32

limit (optional)

Query Parameter — maximum number of records to return format: int32

Return type

array[[User](#)]

Example data

Content-Type: application/json

```
[ {
  "password" : "password",
  "dateJoined" : "2016-08-29T09:12:33.001+0000",
  "userId" : "d290f1ee-6c54-4b01-90e6-d701748f0851",
  "email" : "example@email.com"
}, {
  "password" : "password",
  "dateJoined" : "2016-08-29T09:12:33.001+0000",
  "userId" : "d290f1ee-6c54-4b01-90e6-d701748f0851",
  "email" : "example@email.com"
} ]
```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200
search results matching criteria

400
bad input parameter

Models

[Jump to [Methods](#)]

Table of Contents

1. [Journal -](#)
2. [JournalEntry -](#)
3. [User -](#)

Journal -

[Up](#)

journalId (optional)
[UUID](#) format: uuid
example: d290f1ee-6c54-4b01-90e6-d701748f0851

dateOpened (optional)
[Date](#) format: date-time
example: 2016-08-29T09:12:33.001+0000

consecutiveDays (optional)
[BigDecimal](#)
example: 1.0

JournalEntry -

[Up](#)

entryId (optional)
[UUID](#) format: uuid
example: d290f1ee-6c54-4b01-90e6-d701748f0851

dateCreated (optional)
[Date](#) format: date-time
example: 2021-08-22T01:02:03.001+0000

entryTitle (optional)

[String](#)

entryBody (optional)

[String](#)

User -

[Up](#)

userId

[UUID](#) format: uuid

example: d290f1ee-6c54-4b01-90e6-d701748f0851

email

[String](#) format: email

example: example@email.com

password

[String](#)

dateJoined

[Date](#) format: date-time

example: 2016-08-29T09:12:33.001+0000