Flame Boss API

6/21/17
Version 1.04

Table of Contents

Overview **2**

Versioning **2**

Authentication **3**

Response Format **3**

Request Types **3**

- **Cook Read** **3**
  - Request Parameters 3
  - Response Parameters 3
- **Cooks Index** **4**
- **Cook Update** **4**
- **Cook Delete** **4**
- **Devices Index** **4**
  - Response Parameters 5
- **Signup** **5**
  - Request Parameters 5
  - Response Parameters 5
- **Login** **5**
  - Request Parameters 6
  - Response Parameters 6
- **Logout** **6**
  - Response Parameters 6
- **User Read** **6**
  - Response Parameters 6
- **User Update** **7**
  - Request Parameters 7
- **User Confirm Phone** **7**
  - Response Parameters 7
- **Device Add** **8**
  - Response Parameters 8
- **Device Remove** **8**
  - Response Parameters 8
- **Device Set Set Temp** **8**
  - Response Parameters 8
- **Device Set Pit Alarm** **8**
  - Request Parameters 8
  - Response Parameters 9
- **Device Set Meat Alarm** **9**

Request Parameters 9
Response Parameters 9
Device Set Sound **9**
Request Parameters 9
Response Parameters 9
Device Set Wifi **9**
Request Parameters 10
Response Parameters 10
Device Read **10**
Response Object 10
Apple Push Notification (APN) Support **10**
APN Set **10**
Request Parameters 11
Response Parameters 11
WebSocket API **11**
cook_data **11**
offline **11**
config_changed **12**
new_cook **12**
notice_id **13**

Overview

The Flame Boss API is designed using HTTPS protocol. The client (an app) sends requests to the server and receives responses.

The base URL on the development server is

https://s13.myflameboss.com/api/v1

The base URL on the production server is

https://myflameboss.com/api/v1

Developers must prepend the above strings to the URLs given below.

Versioning

The API version is 2. Every request should include a header as follows:

Name	Value
X-API-VERSION	2

Note that the API version may change independently from the version element of the base URL.

Authentication

The Login request, if successful, returns a token that must be sent along with the username as HTTP headers in any requests that require authentication. The names of these headers are

X-API-USERNAME
X-API-TOKEN

Response Format

Responses are returned in the body of the HTTPS response in JSON format.

Request Types

Following is the description of each request type supported by the API.

Cook Read

Name	Method	URL	Auth?	Parameters	Response
Cook Read	GET	/cooks/<id>	no	skip_cnt max_notice_id	id created_at ended_at title notes online notices data_cnt skip_cnt data

This request returns information about a cook including its cook data for graphing.

Request Parameters

skip_cnt: number of data items to skip in the response because you already have them from a previous request

max_notice_id: the max notice ID you have received already
Response items:

Response Parameters

id: unique ID for the cook

created_at: the date-time the cook started

ended_at: the date-time the cook ended; blank if still in progress

title: title the user has given for the cook

notes: user's notes about the cook in HTML

online: true if the device is online

notices: list of notices to user about the cook

data_cnt: total number of data points stored for the cook

skip_cnt: how many data points were skipped

data: list of data points

Cooks Index

Name	Method	URL	Auth?	Parameters	Response
Cooks Index	GET	/cooks	yes		array of cooks

This request returns a list of all the user's cooks

Cook Update

Name	Method	URL	Auth?	Parameters	Response
Cook Update	PATCH	/cooks/<id>	yes	cook[title] cook[notes]	message

This request allows the user to title the cook, keep the cook permanently, and add notes about the cook.

Cook Delete

Name	Method	URL	Auth?	Parameters	Response
Cook Delete	DELETE	/cooks/<id>	yes		message

Delete cook with ID <id>.

Authenticated user must have the device added in his account.

Devices Index

Name	Method	URL	Auth?	Parameters	Response
Devices Index	GET	/devices	no		username devices

					ip_address ip_devices

This request returns the devices associated with the user according to the user's profile or by IP address.

Devices Index may be authenticated or not. If it is not authenticated the returned username will be an empty string and devices will be an empty as well. Otherwise, these response parameters will contain the username and the devices the user has added.

Response Parameters

ip_address: the IP address of the client
ip_devices: a list of devices that have last connected from ip_address

Signup

Name	Method	URL	Auth?	Parameters	Response
Signup	POST	/registrations	no	registration[username] registration[email] registration[password]	message

This request registers to the user's username, email, and password.

Request Parameters

registration[username]: user's username

registration[email]: user's email address

registration [password]: user's password

Response Parameters

message: OK on success, otherwise an error message

Login

Name	Method	URL	Auth?	Parameters	Response
Login	POST	/sessions	no	session[login] session[password]	user_id username token

This request authenticates the username and password and returns the token to use for authenticated requests. (In CRUD terms this request creates a session.)

Request Parameters

session[login]: user's username or email

session[password]: user's password

Response Parameters

user_id: id of user

username: username of user

auth_token: API token

Logout

Name	Method	URL	Auth?	Parameters	Response
Login	DELETE	/sessions	yes		message

This request logs the user out invalidating the previous API token received. (In CRUD terms this request destroys a session.)

Response Parameters

message: OK on success, otherwise an error message

User Read

Name	Method	URL	Auth?	Parameters	Response
User Read	GET	/registration	yes		id username email terms_of_use terms_accepted_at confirmation_sent_at confirmed_at unconfirmed_email phone texts_enabled text_confirmation_sent_at text_confirmed_at celcius

This request returns the user's own account information

Response Parameters

id: user's user ID

username: user's username
email: user's email
terms_of_use: true or false depending if user accepted the terms of use
confirmation_sent_at: date in the form "2017-02-14T20:11:07.000Z"
confirmed_at: date in the form "2017-02-14T20:11:07.000Z"
phone: user's phone number
texts_enabled: true of false depending if texts are enabled for the user
text_confirmation_sent_at: date in the form "2017-02-14T20:11:07.000Z"
text_confirmed_at: date in the form "2017-02-14T20:11:07.000Z"
celsius: true if celsius, false if fahrenheit

User email is confirmed if confirmed_at is not null.
User phone is confirmed if text_confirmed_at is not null.

User Update

Name	Method	URL	Auth?	Parameters	Response
User Update	PATCH	/registration	Yes		message

This request returns the user's own account information

Request Parameters

registration[id]: user's user ID
registration[username]:
registration[email]:
registration[terms_of_use]: Signifies user agreement to terms and conditions; must be true.
registration[phone]:
registration[texts_enabled]:
registration[celsius]:

User Confirm Phone

Name	Method	URL	Auth?	Parameters	Response
User Confirm Phone	PATCH	/registration/confirm_phone	Yes		message

This request returns the user's own account information

Response Parameters

registration["text_confirmation_code"]: code received by user via text

Device Add

Name	Method	URL	Auth?	Parameters	Response
Device Add	POST	/devices/<id>/add	yes		message

This request adds the device <id> to the user's profile.

Response Parameters

message: OK on success, otherwise an error message

Device Remove

Name	Method	URL	Auth?	Parameters	Response
Device Remove	POST	/devices/<id>/remove	Yes		message

This request removes the device <id> from the user's profile.

Response Parameters

message: OK on success, otherwise an error message

Device Set Set Temp

Name	Method	URL	Auth?	Parameters	Response
Device Set Set Temp	POST	/devices/<id>/set_set_temp	YES	temp_tdc	message

This request changes the set temperature to set_temp (floating point number in Celsius).

temp_tdc: the new set point in tenths degrees Celsius, e.g. 1072 tdc = 107.2 C = 225 F

Response Parameters

message: OK on success, otherwise an error message

Device Set Pit Alarm

Name	Method	URL	Auth?	Parameters	Response
Device Set Pit Alarm	POST	/devices/<id>/set_pit_alarm	yes	en range_tdc	message

Request Parameters

en: 1 to enable, 0 to disable

range_tdc: range away from set point in TDC to trigger alarm

Response Parameters
message: OK on success, otherwise an error message

Device Set Meat Alarm

Name	Method	URL	Auth?	Parameters	Response
Device Set Meat Alarm	POST	/devices/<id>/set_meat_alarm	yes	index mode temp_tdc warm_tdc	message

Request Parameters
index: must be zero
mode: 0 (off), 1 (notice-only), 2 (notice-and-keep-warm)
temp_tdc: the meat temperature that triggers the alarm
warm_tdc: if mode = 2 the set is changed to this temperature when meat alarm triggers

Response Parameters
message: OK on success, otherwise an error message

Device Set Sound

Name	Method	URL	Auth?	Parameters	Response
Device Set Sound	POST	/devices/<id>/set_sound	yes	sound	message

Request Parameters
sound: no sound (0), chirps only (1), chirps and alarms (2)

Response Parameters
message: OK on success, otherwise an error message

Device Set Wifi

Name	Method	URL	Auth?	Parameters	Response
Device Set Wifi	POST	/devices/<id>/set_wifi	yes	ssid sec_type key	message

Request Parameters

ssid: string wifi access point SSID
sec_type: UNSEC (0), WEP (1), WPA (2), WPA2 (3)
key: string wifi password

Response Parameters

message: OK on success, otherwise an error message

Device Read

Name	Method	URL	Auth?	Parameters	Response
Device Read	GET	/devices/<id>	no		device

Response Object

Example:


```
{
 "id": 8838,
 "name": null,
 "online": true,
 "min_set_temp_tdc": 378,
 "max_set_temp_tdc": 2322,
 "most_recent_cook": {
  "id": 62829,
  "device_id": 8838,
  "device_name": null,
  "created_at": "2017-04-26T19:49:14.000Z",
  "ended_at": null,
  "title": null,
  "online": true,
  "notices": [],
  "data_cnt": 9348,
  "skip_cnt": 9348,
  "data": []
 }
}
```


Apple Push Notification (APN) Support

APN Set

Name	Method	URL	Auth?	Parameters	Response
APN Set	POST	/apn_devices	yes	apn_device[device_id] apn_device[token]	message

Request Parameters

apn_device[device_id]: Send notices for this device ID. If zero send notices for all this user's added devices if authenticated. If not authenticated, zero device ID returns an error.

apn_device[token]: The APN token for the Apple mobile device

Response Parameters

message: OK on success, otherwise an error message

WebSocket API

The Flame Boss WebSocket API allows applications to receive new information immediately when it is available on the server without polling the server for updates.

Your WebSocket client must be compatable with websocket-rails.

The WebSocket URL on the development server is

wss://s13.myflameboss.com/websocket

The WebSocket URL on the production server is

wss://myflameboss.com/websocket

After connecting, subscribe to the cook channel for the cook you are monitoring. The name of the channel is simply the cook number as a string. Then your app will receive the following events.

cook_data

Name	Parameters
cook_data	sec, set_temp, pit_temp, meat_temps, fan_dc

sec: Time in Unix epoch format
set_temp: Set temperature in decidegrees Celsius
pit_temp: Pit temperature in decidegrees Celsius
meat_temps: Array of three meat temperatures in decidegrees Celsius
fan_dc: Duty cycle of blower in 100X percentage scale (100% equals 10,000)

offline

Name	Parameters
offline	*None*

The controller has disconnected from the server. There is no separate *online* event because any other message on the channel implies the controller is online.

config_changed

Name	Parameters
config_changed	name, value

The controller reported a change in configuration.

Name	Value
Meat_Alarm_Actions[0]	The meat alarm mode for meat probe 1
Meat_Alarm_Actions[1]	The meat alarm mode for meat probe 2
Meat_Alarm_Actions[2]	The meat alarm mode for meat probe 3
Meat_Alarm_Temps_tdc[0]	The trigger temperature for meat probe 1. In decidegrees Celsius.
Meat_Alarm_Temps_tdc[1]	The trigger temperature for meat probe 2. In decidegrees Celsius.
Meat_Alarm_Temps_tdc[2]	The trigger temperature for meat probe 3. In decidegrees Celsius.
Meat_Alarm_Warm_tdc	The keep-warm temperature that will become the new set temperature when any meat temperature reaches its trigger temperature while the meat alarm mode is Keep Warm. In decidegrees Celsius.
Pit_Alarm_En	The pit alarm enabled mode. Boolean.
Pit_Alarm_Range_tdc	The range about the set temperature such that if the pit temperature exceeds this range the pit alarm will trigger. In decidegrees Celsius.
Sound	The sound setting

Meat_Alarm_Actions use the following codes

0 Off
1 Notice only
2 Notice and Keep Warm

new_cook

Name	Parameters
new_cook	new_cook_id

The current cook ended and a new cook was started with new_cook_id.

notice_id

Name	Parameters
notice_id	notice_id

The controller sent a notice with the given notice_id. Your app must use the HTTPS API to retrieve the notice details. See Cook Read and remember to set max_notice_id to less than notice_id.