



BOTTANGO REST API DOCUMENTATION

Documentation rev 5

End User License Agreement	3
ALL USE OF BOTTANGO IS AT YOUR SOLE RISK.	3
The Bottango REST API	4
What the Bottango REST API is for	4
What can you do with the Bottango REST API	4
How does the Bottango REST work?	4
Bottango REST API Requests and Responses	6
Can Bottango currently animate	6
List of available animations	6
Currently Selected Animation	6
Playback State Details	7
Set Playback State Details	7
Set Input State on a Mapping in a Control Scheme	8
Export Animation Commands	9
Stop All	10

-1-

End User License Agreement

ALL USE OF BOTTANGO IS AT YOUR SOLE RISK.

BOTTANGO IS IN BETA TESTING AND MAY CONTAIN ERRORS, DESIGN FLAWS, BUGS, OR DEFECTS. BOTTANGO SHOULD NOT BE USED, ALONE OR IN PART, IN CONNECTION WITH ANY HAZARDOUS ENVIRONMENTS, SYSTEMS, OR APPLICATIONS; ANY SAFETY RESPONSE SYSTEMS; ANY SAFETY-CRITICAL HARDWARE OR APPLICATIONS; OR ANY APPLICATIONS WHERE THE FAILURE OR MALFUNCTION OF THE BETA SOFTWARE MAY REASONABLY AND FORESEEABLY LEAD TO PERSONAL INJURY, PHYSICAL DAMAGE, OR PROPERTY DAMAGE.

YOU MUST REVIEW AND AGREE TO THE BOTTANGO BETA SOFTWARE END USER LICENSE AGREEMENT BEFORE USING BOTTANGO: <http://www.Bottango.com/EULA>

2-

The Bottango REST API

What the Bottango REST API is for

What exists in this current version of Bottango is just the start of where eventually I hope to take API and scripting control of Bottango. However, just what exists already should be a powerful tool to allow you to use Bottango for even more interesting and diverse use cases. The Bottango REST API allows you to use your own scripts and applications to trigger, control, and get more information on animations in your Bottango project. This could be used to integrate Bottango with your own or other robotic control applications. Or it could be used for you to make a hardware controller to start and stop different animations at a physical button press. If you use this API in an interesting way, let me know! I'd be excited to hear about it!

In the future I hope to add even more scripting support of Bottango, so stay tuned!

Included in this same folder is an example python script for each command. These should show you how to call each function currently available in the REST API.

What can you do with the Bottango REST API

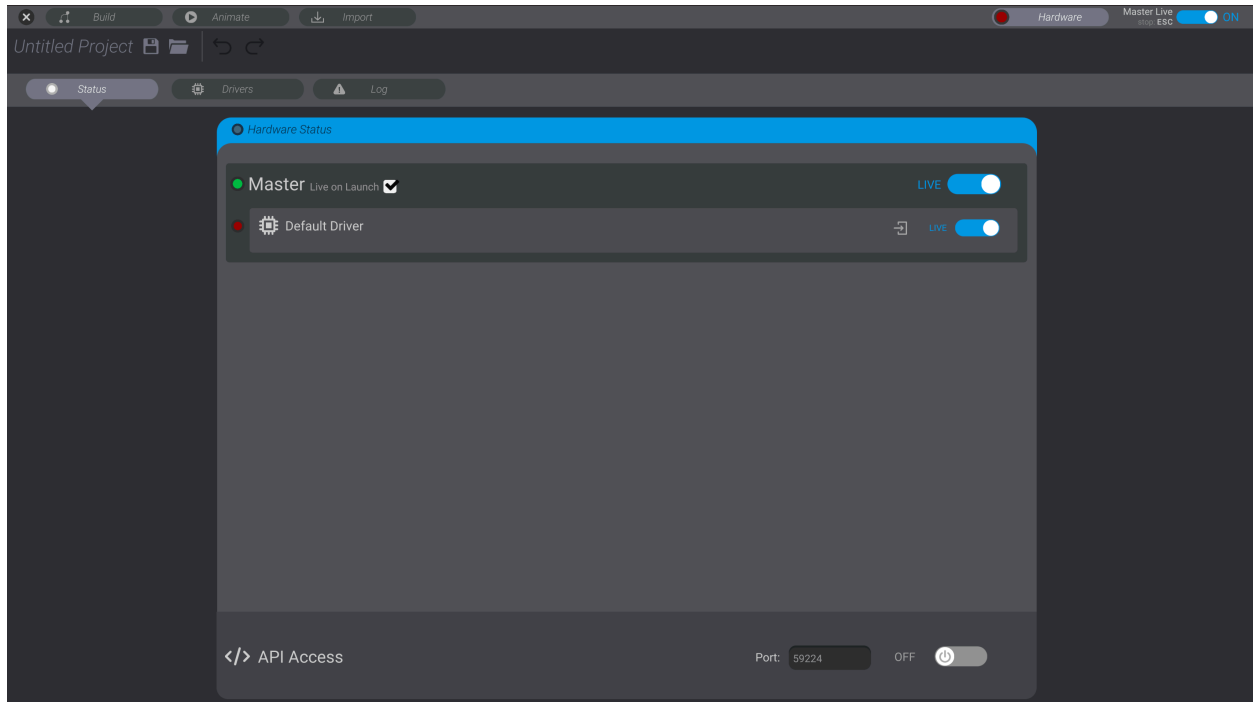
Here is the major functionality available to you currently in the Bottango REST API:

- Get whether Bottango is currently able to start playing an Animation.
- Get information on the animations available in your project, and which animation is currently selected.
- Get information on the current playback state of Bottango, including the signal and movement of all effectors in the project.
- Export animation commands to JSON
- Start / stop playing an animation.
- Start / stop recording live controller input (if a control scheme is selected)
- Set playback time.
- Change the selected animation.
- Drive input into the input recording features of Bottango
- Call "STOP" in Bottango (the same as pressing the escape key on the keyboard).

How does the Bottango REST work?

The Bottango REST API works as a local server on your machine. You call into it with GET and PUT requests the way you would a remote REST API. If you can make and receive JSON in a web request, you can interact with the Bottango REST API.

You can enable the Bottango REST API in a project by toggling the “API Access” toggle in the hardware menu:



Once it is enabled, you will see an “API” notification in the top bar as well:



Connection to the Bottango REST API is by default as a local server on port 59224 (<http://localhost:59224/>). You can change the port if needed in that same menu. In theory, if you are able to resolve the port forwarding on your local network, you could access the API from remote machines as well.

See the example python script mentioned above to see an example of connecting and sending requests to the API in python. For other languages, the same lessons will apply.

3-

Bottango REST API Requests and Responses

Can Bottango currently animate

URI: /CanAnimate/ GET

Request Paramaters:

None

Response Parameters:

Bool canAnimate

Returns the current state of Bottango, if Bottango could begin or could potentially already be playing animations. Useful to check before making other calls if you're not sure the state of the Bottango application, as if Bottango is not able to animate, some of the other subsequent calls will return an error.

List of available animations

URI: /Animations/ GET

Request Paramaters:

None

Response Parameters:

string[] animations

Returns the list of animations in the project. The order of animations in the array is also the order you can use for calling change animation if you want to use an index instead of a name.

Currently Selected Animation

URI: /Animations/Selected/ GET

Request Paramaters:

None

Response Parameters:

string selectedAnimationName

int selectedAnimationIndex

Returns the name and index of the currently selected animation. This is the animation that would play if you told Bottango to start playing (or that is playing now if Bottango is already playing).

Playback State Details

URI: /PlaybackState/ GET

Request Paramaters:

None

Response Parameters:

```
int selectedIndex
string selectedAnimationName
bool isPlaying
int playbackTimeInMS
int durationInMS
obj[] effectors
    string name
    bool live
    string identifier
    string driverName
    bool driverLive
    float movement
    float signal
```

Returns details on the playback state of Bottango at the moment the API call is made. This includes the selected animation as called above, as well if Bottango is currently playing now, the position of the playhead, and the duration of the currently selected clip (both in milliseconds)

As well, the call returns an array of objects, each representing an effector in the Bottango project. This is the exact state of each effector in the project at the time of the API call. Besides basic effector state (like live, identifier, driver name, etc.) you also will get the movement and signal value of the effector. Movement is the signal value normalized to 0 - 1.

Set Playback State Details

URI: /PlaybackState/ PUT

Request Paramaters:

```
int selectedIndex
string selectedAnimationName
bool isPlaying
bool isRecording
int playbackTimeInMS
```

Response Parameters:

None

Sets the playback state of Bottango. You can pick and choose which elements of the playback state you want to control in the call, and add only those parameters to the request you wish to change or control.

Depending on the parameters you have in the request, and the current state of Bottango's playback, some of the requested changes may happen in a sequence in order to fully resolve the request.

Changing Selected Animation

The selected animation index and selected animation name parameter fill the same purpose. If both are in the request, only the index will be used. If you use name, and multiple animations in the project have the same name, the call will return an error.

If you call to change the selected animation while playing, Bottango will stop playing and then switch. This can be overridden by including `isPlaying` to `True` in the request.

If you're not sure what animation has which index, use the `/Animations/` call above to get an ordered list of all animations in the project.

Changing Playhead Time

If you change the playhead time while Bottango is currently playing, Bottango will stop playing and then jog to the requested time. This can be overridden by including `isPlaying` to `True` in the request, in which case Bottango will jog to the requested time, and then begin playing again.

If you include this parameter in a request that also changes the selected animation, Bottango will use the requested time as the starting time of the requested animation, instead of the default behavior which is the current time.

Starting / Stopping Play

On its own, this parameter acts the same as pressing start or stop. When combined with changing the selected animation and/or the playhead time, setting this parameter to `true` will override the default behavior of stopping an existing playback on switching time/animation, and instead make the switch then resume playing.

Starting / Stopping Recording

This follows the same behavior as starting / stopping play, but will result in a recording session beginning instead of a play session if set to `true`. If you set the value of `isRecording`, any value on `isPlaying` will be ignored.

Set Input State on a Mapping in a Control Scheme

URI: `/ControllInput/` PUT

Request Paramaters:

string identifier
float value

Response Parameters:

None

Sets the input value of a mapping in a control scheme that has had it's input type set to "Rest API."

In order to not require you to carefully monitor when you stream out data to be recorded, this API call is fairly permissive, and will not return an error if it receives an unknown identifier or receives data while not actively recorded. However, a warning will be added to the hardware log for each set input request that Bottango is unable to process when received. If you're not seeing input come in the way you would expect, check the log and see if Bottango was unable to process the request.

Each request requires both an identifier and a value. When you create a mapping in a control scheme, and select "Rest API" as the input type, in the configuration options for that mapping, you can enter a string identifier in the Bottango app. This defaults to "myIdentifier" but you can use any unique string you'd like for each mapping. Collided identifiers are allowed, and if multiple mappings have the same identifier, each will be controlled by the incoming value independently.

Value should be a float between 0.0 and 1.0. Any value outside of that range or non float value will return an error. The parsing of the float value depends on the kind of part being controlled:

- Movement based parts (joints, motors, an axis of a pose blends, a color channel of a color event, etc) will 1:1 map the float to the expected movement of the part. 0.0 being the minimum range of motion and 1.0 the maximum.
- On Off events will treat any non 0.0 value that is less than 1.0 as on, and will treat 0.0 as off.
- Trigger events will treat any non 0.0 value that is less than 1.0 as a trigger event for each request received. 0.0 will be ignored.

You will only see the result of this control while actively recording (or in snapshot recording). While not recording, Bottango will ignore these requests, but not return an error as discussed above. After recording is complete, the collected input requests will be converted into keyframes on the animation timeline, the same as input from a gamepad, etc.

Export Animation Commands

URI: /Animations/AnimationCommands/ GET

Request Parameters:

int[] animationIndexes

string[] animationNames

int startingAnimationIndex

String startingAnimationName

object[] effectorIdentifiers

Response Parameters:

JSON formatted animation commands

Exports out the command strings of all animations in the project, following the same ideas as the "Export to Code" button in the animation view of Bottango.

Animation Indexes/names are optional parameters, that set which animations to include in the export. If unset, the response will include all animations. If both are received, indexes will be used.

Starting animation name/index is optional and used to determine which animation is used for the starting motor positions in the register commands of the exported commands. If both are received, index will be used. If unset, the response will use animation index 0.

Effector identifiers is an optional array of objects to include only certain drivers and/or effectors in the response. If left unset, all drivers and effectors will be include. The format of the request array of objects should look like this:

```
[
  { "Name" : "Default Driver", "Identifiers" : [3, 5, 6]},
  { "Name" : "My Other Driver", "Identifiers" : [3, 5, 6]},
]
```

Each object should have a 'Name' field, with the name of the driver as a string and an 'Identifiers' field which is an array of strings, each string the identifier of a desired effector to include.

Stop All

URI: /Stop/ GET

Request Paramaters:

None

Response Parameters:

None

Stops all motors, and sets all drivers and master to not live. This is the same as pressing the escape key on your keyboard in Bottango. By design there is no opposite command to set things live. You should intentionally set motors, drivers, and master live in the Bottango application, and use this as a quick stop when needed.