



## **BOTTANGO PLAYBACK API DOCUMENTATION**

Documentation rev 3

<b>End User License Agreement</b>	<b>3</b>
ALL USE OF BOTTANGO IS AT YOUR SOLE RISK.	3
<b>The Bottango Playback API</b>	<b>4</b>
What the Bottango Playback API is for	4
What can you do with the Bottango Playback API	4
How does the Bottango Playback API work?	4
<b>Bottango Playback API Requests and Responses</b>	<b>6</b>
Can Bottango currently animate	6
List of available animations	6
Currently Selected Animation	6
Playback State Details	7
Set Playback State Details	7
Stop All	8

-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 Playback API*

### *What the Bottango Playback 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 Playback 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 "PlaybackAPIExample.py." This is a python script with examples of how to call each function currently available in the Bottango Playback API.

### *What can you do with the Bottango Playback API*

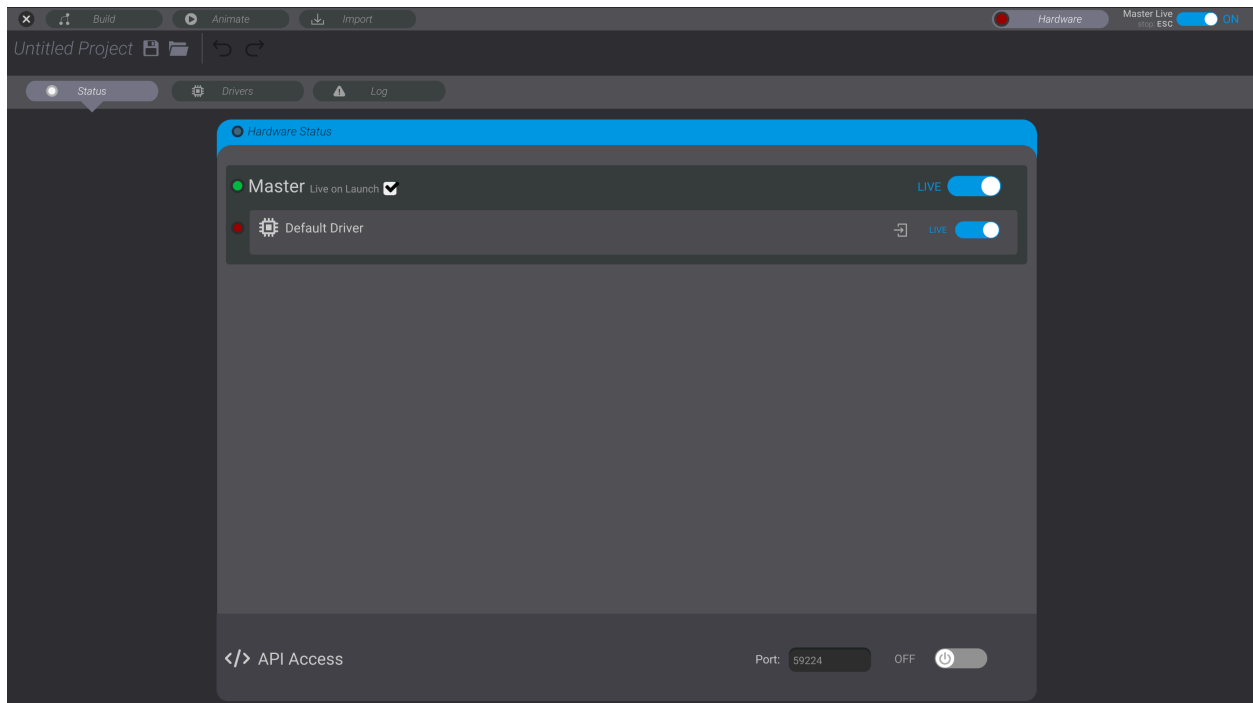
Here is the major functionality available to you currently in the Bottango Playback API:

- 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.
- Start / stop playing an animation.
- Start / stop recording live controller input (if a control scheme is selected)
- Set playback time.
- Change the selected animation.
- Call "STOP" in Bottango (the same as pressing the escape key on the keyboard).

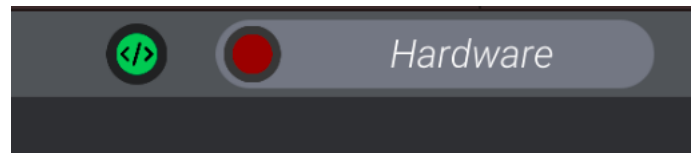
### *How does the Bottango Playback API work?*

The Bottango Playback 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 Playback API.

You can enable the Bottango Playback 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 Playback 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 Playback 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.

## 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.