Setting on Webhook.site

1. Create a new URL

My URLs

RMIT CODE • dlt.rmit.edu.vn@gmail.com

+ New URL   All URLs (1) ⌄   New group...   Delete (0)

Create New URL   ✕

Customize how the URL will respond. You can also use Custom Actions to change the response dynamically.

Default status code	200
Content Type	text/plain
Timeout before response	2000 ⬍
Content ⓥ	

Variables ▲

Create

2. Go back to Shopify webhook settings and paste the URL

3.   Turn on Custom Actions and Settings

              In this example, I'll extract the item name, and deliver it to different zones. Depends on your scenario, you might want to extract different
              data (customer address, quantity, …)
4.   Using custom actions to extract the data from webhook JSON, save it to a custom variable.
5.   Make sure the data in the JSON is:

              "line_items": [
              {   "title": "Studio Bag",}

      Otherwise you need to change the JSONPath in custom actions below:

Get item ID from Shopify

6.   Get itemID Quantity from Shopify

#1 Extract JSONPath

Variable name: $product_title$
JSONPath: .line_items[0].name

#2 Extract JSONPath

Variable name: $product_quantity$
JSONPath: .line_items[0].quantity

#3 Extract JSONPath

Variable name: $product_title2$
JSONPath: .line_items[1].name

#4 Extract JSONPath

Variable name: $product_quantity2$
JSONPath: .line_items[1].quantity

#5 Extract JSONPath

Variable name: $product_quantity3$
JSONPath: .line_items[2].quantity

#6 Extract JSONPath

Variable name: $product_title3$
JSONPath: .line_items[2].name

#7 Extract JSONPath

Variable name: $product_title4$
JSONPath: .line_items[3].name

← Back   Editing Action #2 Extract JSONPath

Source Ⓥ

$request.content$

The source text used for the JSONPath extraction. If empty, the request content is used.

JSONPath *

.line_items[0].quantity

Get part of the JSON content with JSONPath and store the first result in the variable. If there are multiple results, they are comma separated. JSONPath examples.

Default Value Ⓥ

0

If the value can't be found, the variable will be set to this instead.

Variable Name *

$ product_quantity $

The variable can be used in any action after this one (which makes the action order important.)

☐ Repeat Beta

When checked, actions after this are repeated for each item in the resulting output of the JSONPath query.

Only run when condition passes

Always run ▼

☐ Queue ❶   ☐ Disable

▼ Outputs
▶ Response
▶ Variables

Test   Save Action *

↑ ↓   ＋   Variables ▲   Import...   Export...   Close

7. Similar for item ID2 item ID3

Custom Actions

#1 Extract JSONPath
Variable name: $product_title$
JSONPath: .line_items[0].name

#2 Extract JSONPath
Variable name: $product_quantity$
JSONPath: .line_items[0].quantity

#3 Extract JSONPath
Variable name: $product_title2$
JSONPath: .line_items[1].name

#4 Extract JSONPath
Variable name: $product_quantity2$
JSONPath: .line_items[1].quantity

#5 Extract JSONPath
Variable name: $product_quantity3$
JSONPath: .line_items[2].quantity

#6 Extract JSONPath
Variable name: $product_title3$
JSONPath: .line_items[2].name

#7 Extract JSONPath
Variable name: $product_title4$
JSONPath: .line_items[3].name

← Back   Editing Action #3 Extract JSONPath

Source ⓥ

$request.content$

The source text used for the JSONPath extraction. If empty, the request content is used.

JSONPath *

.line_items[1].name

Get part of the JSON content with JSONPath and store the first result in the variable. If there are multiple results, they are comma separated. JSONPath examples.

Default Value ⓥ

If the value can't be found, the variable will be set to this instead.

Variable Name *

$ product_title2 $

☐ Repeat Beta

When checked, actions after this are repeated for each item in the resulting output of the JSONPath query.

The variable can be used in any action after this one (which makes the action order important.)

Only run when condition passes

Always run   ▾

☐ Queue ⓘ   ☐ Disable

▾ Outputs
▸ Response
▸ Variables

Test

Save Action *

↑ ↓   ✚   Variables ▲   Import...   Export...   Close

8.   Final script to request URL, change the api token accordingly as your Blynk token.
Each item will request to change value for a different virtual pin.

Custom Actions

← Back   Editing Action #9 WebhookScript

#3 Extract JSONPath

Variable name: $product_title2$
JSONPath: .line_items[1].name

#4 Extract JSONPath

Variable name: $product_quantity2$
JSONPath: .line_items[1].quantity

#5 Extract JSONPath

Variable name: $product_quantity3$
JSONPath: .line_items[2].quantity

#6 Extract JSONPath

Variable name: $product_title3$
JSONPath: .line_items[2].name

#7 Extract JSONPath

Variable name: $product_title4$
JSONPath: .line_items[3].name

#8 Extract JSONPath

Variable name: $product_quantity4$
JSONPath: .line_items[3].quantity

#9 WebhookScript
ID: 0874c

Script editor   Docs, examples, shortcut keys ☑   Built-in functions ☑

```
1  theURL = "https://sgp1.blynk.cloud/external/api/update?token=AC74vSrGWlko7hDdCD_xNN2qJkg2FLWS&"
2
3  if (var('product_title') == "Studio Bag"){
4      request(theURL + "v1=" + var('product_quantity').to_string(), method='GET')
5  }
6
7  if (var('product_title') == "Cumulus"){
8      request(theURL + "v3=" + var('product_quantity').to_string(), method='GET')
9  }
10
11 if (var('product_title') == ("Dance Bag Nylon - Black")){
12     request(theURL + "v4=" + var('product_quantity').to_string(), method='GET')
13 }
14
15 if (var('product_title') == ("Dance Bag Nylon - Pink")){
16     request(theURL + "v4=" + var('product_quantity').to_string(), method='GET')
17 }
18
19 if (var('product_title') == ("Stratus Backpack - Black")){
20     request(theURL + "v2=" + var('product_quantity').to_string(), method='GET')
21 }
```

Only run when condition passes

Always run ▼

☐ Queue ❶   ☐ Disable

▼ Outputs
▶ Response
▶ Variables

Test   Save Action *

↑   ↓   +   Variables ▲   Import...   Export...   Close

The code for request call will be left at webhook.site script.js