



Submitting an open-source project to OSHWA

Created by Dylan Herrada



Last updated on 2020-11-25 12:30:30 PM EST

Overview



OSHW (Open Source Hardware Association) is an organization that, among other things, has created and maintains a process to allow users and companies to certify that their projects are open source. This past spring, many of the boards that Adafruit manufactures were certified. Those certifications were submitted semi-manually, which took quite a while. Now, OSHWA has [an API \(https://adafru.it/ONa\)](https://adafru.it/ONa) that allows you to submit projects much quicker and can make submitting large numbers of projects at the same time much easier and more straightforward.

In this guide, you'll learn how to submit your open-source project for certification using this API and a Python script. It'll take you through the process used by Adafruit to submit a project.

API Docs

[API Docs \(https://adafru.it/OFn\)](https://adafru.it/OFn)

Using the script

Before you start, you'll need to make an API key that you'll need for use in the script. You can do that here:

<https://adafru.it/ONb>

<https://adafru.it/ONb>

For the Python code, download the project zip below, and then unzip it in a folder of your choice. **Make sure not to run it until you've edited the file to fill out all of the fields with your project's information.**

```

import requests

token = "TOKEN_GOES_HERE"

url = "https://certificationapi.oshwa.org/api/projects"

responsiblePartyType = "Company"
responsibleParty = "Adafruit Industries, LLC"
bindingParty = "Limor Fried"
country = "United States of America"
streetAddress1 = "150 Varick St."
streetAddress2 = ""
city = "New York"
state = "New York"
postalCode = "10013"
privateContact = "oshw@adafruit.com"
publicContact = "oshw@adafruit.com"
projectName = "Adafruit QT Py = SAMD21 Dev Board with STEMMA QT"
projectWebsite = "https://www.adafruit.com/product/4600"
projectVersion = "Rev C"
previousVersions = []
projectDescription = "This diminutive dev board comes with our favorite lil chip, the SAMD21. OLEDs! Inertial Measurement Units! Sensors a-plenty. All plug-and-play thanks to the innovative chainable design: SparkFun Qwiic-compatible STEMMA QT connectors for the I2C bus so you don't even need to solder. It also has 11 GPIO pins and a built-in NeoPixel RGB LED. This board ships with CircuitPython but also works great with Arduino."
primaryType = "Electronics"
additionalType = ['Electronics']
projectKeywords = f'["{projectName}", "CircuitPython"]'
documentationUrl = "https://learn.adafruit.com/adafruit-qt-py"
hardwareLicense = "Other"
softwareLicense = "MIT"
documentationLicense = "CC BY-SA"
relationship = "PROJECT IS DESIGNED AND DISTRIBUTED BY ADAFRUIT INDUSTRIES"

payload = f'{{"responsiblePartyType": "{responsiblePartyType}", "responsibleParty": "{responsibleParty}", "bindingParty": "{bindingParty}", "country": "{country}", "streetAddress1": "{streetAddress1}", "streetAddress2": "{streetAddress2}", "city": "{city}", "state": "{state}", "postalCode": "{postalCode}", "privateContact": "{privateContact}", "publicContact": "{publicContact}", "projectName": "{projectName}", "projectWebsite": "{projectWebsite}", "projectVersion": "{projectVersion}", "previousVersions": {previousVersions}, "projectDescription": "{projectDescription}", "primaryType": "{primaryType}", "additionalType": {additionalType}, "projectKeywords": {projectKeywords}, "citations": [{}], "documentationUrl": "{documentationUrl}", "availableFileFormat": true, "hardwareLicense": "{hardwareLicense}", "softwareLicense": "{softwareLicense}", "documentationLicense": "{documentationLicense}", "noCommercialRestriction": true, "noDocumentationRestriction": true, "openHardwareComponents": "true", "creatorContribution": true, "noUseRestriction": true, "redistributedWork": true, "noSpecificProduct": "true", "noComponentRestriction": true, "technologyNeutral": true, "certificationMarkTerms": {{ "accurateContactInformation": {{ "term": "I have provided OSHWA with accurate contact information, recognize that all official communications from OSHWA will be directed to that contact information, and will update that contact information as necessary.", "agreement": true }}}, "explanationCertificationTerms": "N/A (I agree to all terms)", "relationship": "{relationship}", "agreementTerms": true, "parentName": ""}}}'
print(payload)

headers = {"Content-Type": "application/json", "Authorization": f"Bearer {token}"}

r = requests.request("POST", url, headers=headers, data=payload)

print("\n")
print(r.status_code)
print("\n")
print(r.json())

```

Then, fill out the variables in the script above. Some are pretty obvious, but some can be slightly more

confusing. They're briefly explained below. If you're still uncertain as to the data you need for a field, visit the [webpage where you can certify a project \(https://adafruit.com/submitting-an-open-source-project-to-oshwa\)](https://adafruit.com/submitting-an-open-source-project-to-oshwa) and find the field that matches the uncertain variable.

token - The API token you got from the link above.

url - The URL to POST to. Don't change this.

responsiblePartyType - Who this certification is on behalf of, can be "Individual", "Company", or "Organization".

responsibleParty - The name of who or what is responsible for the item being certified.

bindingParty - If responsiblePartyType isn't "Individual", this field should contain the name of the person who is giving permission to OSHWA to certify the project.

country - The country the address you'd like OSHWA to have on file is in.

streetAddress1 - Street address line 1.

streetAddress2 - Street address line 2.

city - City.

state - State.

postalCode - Postal code.

privateContact - The email address you'd like OSHWA to use to contact you.

publicContact - The email address you'd like the public to contact you about your open-source project.

projectName - The name of the project.

projectWebsite - Your project's website.

projectVersion - Your project's current version.

previousVersions - Put the UID for any previously registered versions of the project in this list.

projectDescription - Your project's description. Should be under 500 characters. If you need to put a double quote in here, use the format `\"\"`.

primaryType - Primary project type. See the certification form webpage for more info on possible values.

additionalType - Additional project types, should be formatted similarly to a Python array.

projectKeywords - List of project keywords. Adafruit uses the project name as the first one, although this is by no means necessary.

documentationUrl - URL of the documentation.

hardwareLicense - Hardware license type. Options are "CERN", "Solderpad", "TAPR", and "Other".

softwareLicense - Software license type.

documentationLicense - Documentation license type. Options are "CC 0", "CC BY", "CC BY-SA", and "Other".

relationship - The relationship of the **responsibleParty** to the project.

After you've filled everything out, you'll want to run the script with **python3 submitter.py**

```
python3 submitter.py
{"responsiblePartyType": "Company", "responsibleParty": "Adafruit Industries, LLC", "bindingParty": "Linar Fried", "country": "United States of America", "streetAddress": "150 Varick St.", "streetAddress2": "150 Varick St.", "city": "New York", "state": "New York", "postalCode": "10013", "privateContact": "oshw@adafruit.com", "publicContact": "oshw@adafruit.com", "projectName": "Adafruit MagTag - 2.9\" GrayScale E-Ink WFL Display", "projectWebsite": "https://www.adafruit.com/product/4806", "projectVersion": "New C", "previousVersions": [], "projectDescription": "The Adafruit MagTag combines the new ESP32-S2 wireless module and a 2.9\" grayscale E-ink display to make a low-power IoT display that can show data on its screen even when power is removed. The ESP32-S2 is great because it builds on the years of code and support for the ESP32 and also adds native USB support so you can use this board with Arduino or CircuitPython! We designed this board to be low-power friendly. During deep sleep (~250uA power draw), you can run for a few weeks between charges.", "primaryType": "Electronics", "additionalType": ["Electronics"], "projectKeywords": ["Adafruit MagTag - 2.9\" GrayScale E-Ink WFL Display", "CircuitPython"], "citations": [], "documentationUrl": "https://github.com/adafruit/Adafruit_MagTag_PCB_s", "availabilityFormat": true, "hardwareLicense": "Other", "softwareLicense": "MIT", "documentationLicense": "CC BY-SA", "noCommercialRestriction": true, "noDocumentationRestriction": true, "noOpenHardwareComponents": true, "creatorContribution": true, "noRestriction": true, "redistributedWork": true, "noSpecificProduct": true, "noComponentRestriction": true, "technologyNeutral": true, "certificationTerms": [{"accurateContactInformation": [{"term": "I have provided OSHWA with accurate contact information, recognize that all official communications from OSHWA will be directed to that contact information, and will update that contact information as necessary."}], "agreement": true}], "explanationCertificationTerms": "N/A (I agree to all terms)", "relationship": "PROJECT IS DESIGNED AND DISTRIBUTED BY ADAFRUIT INDUSTRIES", "agreementTerms": true, "parentName": ""}

200
{"success": true, "project": {"responsiblePartyType": ["en-US": "Company"], "responsibleParty": ["en-US": "Adafruit Industries, LLC"], "bindingParty": ["en-US": "Linar Fried"], "country": ["en-US": "United States of America"], "streetAddress": ["en-US": "150 Varick St."], "streetAddress2": ["en-US": "150 Varick St."], "city": ["en-US": "New York"], "state": ["en-US": "New York"], "postalCode": ["en-US": "10013"], "privateContact": ["en-US": "oshw@adafruit.com"], "publicContact": ["en-US": "oshw@adafruit.com"], "projectName": ["en-US": "Adafruit MagTag - 2.9\" GrayScale E-Ink WFL Display"], "projectWebsite": ["en-US": "https://www.adafruit.com/product/4806"], "projectVersion": ["en-US": "New C"], "projectDescription": ["en-US": "The Adafruit MagTag combines the new ESP32-S2 wireless module and a 2.9\" grayscale E-ink display to make a low-power IoT display that can show data on its screen even when power is removed! The ESP32-S2 is great because it builds on the years of code and support for the ESP32 and also adds native USB support so you can use this board with Arduino or CircuitPython! We designed this board to be low-power friendly. During deep sleep (~250uA power draw), you can run for a few weeks between charges."], "primaryType": ["en-US": "Electronics"], "additionalType": ["en-US": ["Electronics"]], "projectKeywords": ["en-US": ["Adafruit MagTag - 2.9\" GrayScale E-Ink WFL Display", "CircuitPython"]], "citations": ["en-US": []], "documentationUrl": ["en-US": "https://github.com/adafruit/Adafruit_MagTag_PCB_s"], "availabilityFormat": ["en-US": true], "hardwareLicense": ["en-US": "Other"], "softwareLicense": ["en-US": "MIT"], "documentationLicense": ["en-US": "CC BY-SA"], "noCommercialRestriction": ["en-US": true], "noDocumentationRestriction": ["en-US": true], "openHardwareComponents": ["en-US": true], "creatorContribution": ["en-US": true], "noRestriction": ["en-US": true], "redistributedWork": ["en-US": true], "noSpecificProduct": ["en-US": true], "noComponentRestriction": ["en-US": true], "technologyNeutral": ["en-US": true], "certificationTerms": ["en-US": [{"accurateContactInformation": [{"term": "I have provided OSHWA with accurate contact information, recognize that all official communications from OSHWA will be directed to that contact information, and will update that contact information as necessary."}], "agreement": true}], "explanationCertificationTerms": ["en-US": "N/A (I agree to all terms)"], "relationship": ["en-US": "PROJECT IS DESIGNED AND DISTRIBUTED BY ADAFRUIT INDUSTRIES"], "agreementTerms": ["en-US": true]}}
```

If it was successful, you should see a 200 on the second line that gets printed. It should look something like this. If it doesn't look like this, then it is recommended that you read [this section of OSHWA's API docs](https://adafru.it/OFn) (<https://adafru.it/OFn>).

