How to use the SurveyMonkey API

Yidan Wang [Follow]

May 29, 2018 · 5 min read ★

SurveyMonkey is one of the most popular survey design and distribution tool in the market. It has some robust features and an intuitive interface. But it can be tedious to manually create various collectors for a survey. Lots of the steps are repetitive, and currently there is no clone functionality.

We can leverage SurveyMonkey's APIs to automate the collector creation process and much more.

In this demo, I will write a simple Google App Script function that creates a collector for a SurveyMonkey survey automatically.

First, you need to create a Google App Script project. Go to Google Drive -> New -> More -> Google App Script.

In order to make a request to SurveyMonkey through our script, we need to use a Google App Script service called URL Fetch Service.
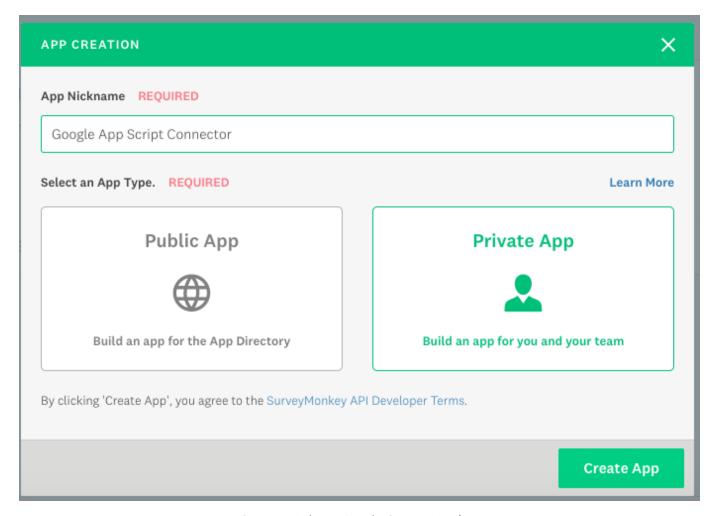
We will create a POST request to SurveyMonkey to create a collector for a survey, according to the Survey Monkey API.

Before we get into the details of how to create the request, we need to authorize our script to make changes on behalf of our SurveyMonkey account. To do so, create an App in SurveyMonkey. This will give us an Access Token to pass along in our request.

There are two types of apps in SurveyMonkey — public and private. For our purpose, a private app is sufficient. But do note, it's subject to API call limit: 120 calls/minute and 500 calls/day. Additional calls are available for purchase. If you need to create a more
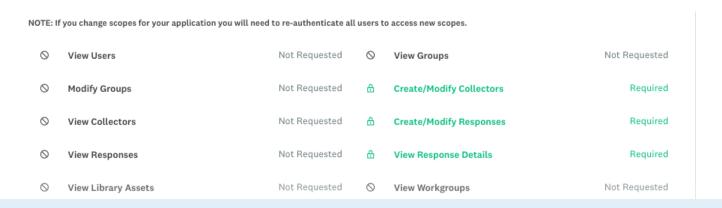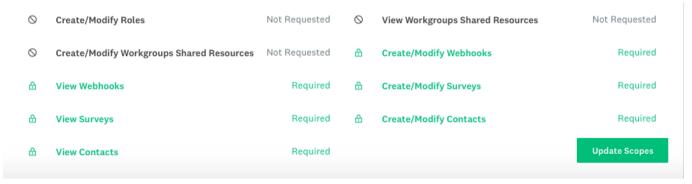
How to Get the Access Token

1. Login to your SurveyMonkey account at https://developer.surveymonkey.com/ and create a Private App.

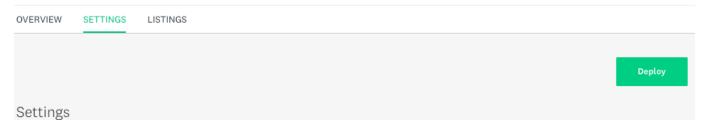Create a Private App in SurveyMonkey

2. Go to Settings -> Update Scopes ->Deploy the App

This is your last free story this month. Sign up and get an extra one for free.   ✕

⊘	Create/Modify Roles	Not Requested	⊘ View Workgroups Shared Resources	Not Requested
⊘	Create/Modify Workgroups Shared Resources	Not Requested	🔒 Create/Modify Webhooks	Required
🔒	View Webhooks	Required	🔒 Create/Modify Surveys	Required
🔒	View Surveys	Required	🔒 Create/Modify Contacts	Required
🔒	View Contacts	Required	Update Scopes	

Update Scopes required for the app

Google App Script Connector

OVERVIEW   SETTINGS   LISTINGS

Deploy

Settings

Deploy the App by hitting the 'Deploy' Button

3. Get the Access Token under the Credentials section in the Settings tab

Credentials   🔍 REFRESH CLIENT

Client ID
ziuP-bnV▨▨▨▨▨▨▨

Secret
•••••••••••   👁

Access Token
f-i4▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨

Quota Limits
500 / day : 120 / minute

Daily Quota Usage   0 / 500

Access Token is the in the settings tab

On the side note, if you change the scopes for your app in the future, it will generate a
new Access Token.

For the purpose of this project, we will only need to access our own SurveyMonkey account, so this Access Token is sufficient.

If your app will access many SurveyMonkey accounts, implement the OAuth 2.0 three-step flow. This flow generates a long-lived access token your app can use with every API call to the associated SurveyMonkey account. It's important to note the access token only grants access when used in combination with your API credentials (client ID) and only to the SurveyMonkey account which was authorized. Your app will need to obtain additional access tokens for each SurveyMonkey account you wish to access.

Authorize Our Script With the Access Token

Now we have our Access Token! We need to pass along the access token when we try to access SurveyMonkey through our script. To do so, we need to pass header information for each request we make in the following format:

'headers':{'Authorization': 'bearer your_access_token'}

Before we make a collector, we need one more piece of information — the ID of the survey we will create a collector for. To my knowledge, this ID is not accessible in the SurveyMonkey interface. But we can find it through the API.

Find the Survey ID through API

Write the following function in your Google App Script file.

function getSurveyID(){

var options = {
'method' : 'GET',
'headers':{'Authorization':'bearer your_access_token'},
'contentType': 'application/json'
};
var result = UrlFetchApp.fetch('https://api.surveymonkey.com/v3/surveys?title=Your%20Survey', options);

```
var params = JSON.parse(result.getContentText());

Logger.log(params.data[0].id);

}
}
```

Replace your_access_token with your access token, Your%20Survey with the title of your survey.

Then save your script, go to the top of the script file, select getSurveyID from the dropdown list and click on the start button.

Make a getSurveyID function call

This will make a GET request to SurveyMonkey and log the ID of the first survey with the same title.

Go to View -> Logs. Save that ID!

Congrats! You've just made your first SurveyMonkey API call! If you go to your App's dashboard, you will see your API call number has increased.

Create An Collector for the Survey

Now we are ready to auto-create our collector! According to the SurveyMonkey API, we need to post a POST request to the following URL: `https://api.surveymonkey.com/v3/surveys/{survey_id}/collectors` . Replace {survey_id} with the ID you obtained in the previous step.

Another required parameter is type. You need to specify if the collector is 'weblink' or 'email'. All other parameters are optional. Of course, you would also need to pass along the access token for authorization.

This is your last free story this month. Sign up and get an extra one for free.   ✕

```
function createCollector(){

// Make a POST request with a JSON payload.
var data = {
"type": "weblink"
};

var options = {
'method' : 'post',
'headers':{'Authorization':'bearer your_access_token'},
'contentType': 'application/json',
// Convert the JavaScript object to a JSON string.
'payload' : JSON.stringify(data),


};


var response =
UrlFetchApp.fetch('https://api.surveymonkey.com/v3/surveys/{survey_id}/collectors',
options);
Logger.log(response.getContentText());

}
```

Once you run this function, you've successfully created a new collector for your survey!
Hooray! Go to your Log, you can see the HTTP Response sent by SurveyMonkey. It
contains details about the collector such as id, title and close date.

You may also specify the details when creating the Collector by passing along the
optional parameters listed in the API.

For intance:

var data = {

"thank_you_message":"Thank you for completing this survey!",

"close_date":"2018–05–30T22:10:00+00:00",

"redirect_url":"https://google.com"

};

Do pay attention to the data type required. If you pass along a wrong format, it will throw an invalid schema error at you.

The most tricky one is the Date data type for "close_date". As specified in the API, dates are usually in the format YYYY-MM-DDTHH:MM:SS+HH:MM. All date strings are implicitly in UTC.

Going Beyond

Congrats on completing your first successful SurveyMonkey collector automation! Now you can leverage the power of automation to make your survey distribution painless. For instance, you can modify the script to create 100 collectors with the same naming pattern in a single click! You can automatically distribute the survey to the staff in a department when the department head submits a request form.

No more manual work thanks to automation!

API   Surveys   Surveymonkey   Google Apps Script   Automation

Medium

About   Help   Legal

Get the Medium app