Development Release Notes

Openbadges Activity Rule Events: Google Sheets Add-on Development release notes

[Version number]

Released to Dev	[Date]
Released to Testing	[Date]
Released To Demo	[Date]
Released To Live	[Date]

MYKNOWLEDGEMAP®

Table of Contents

Introduction

This document contains the release notes for *Openbadges Activity Rule Events - Google Sheets and Google Forms Add-on*.

What's New

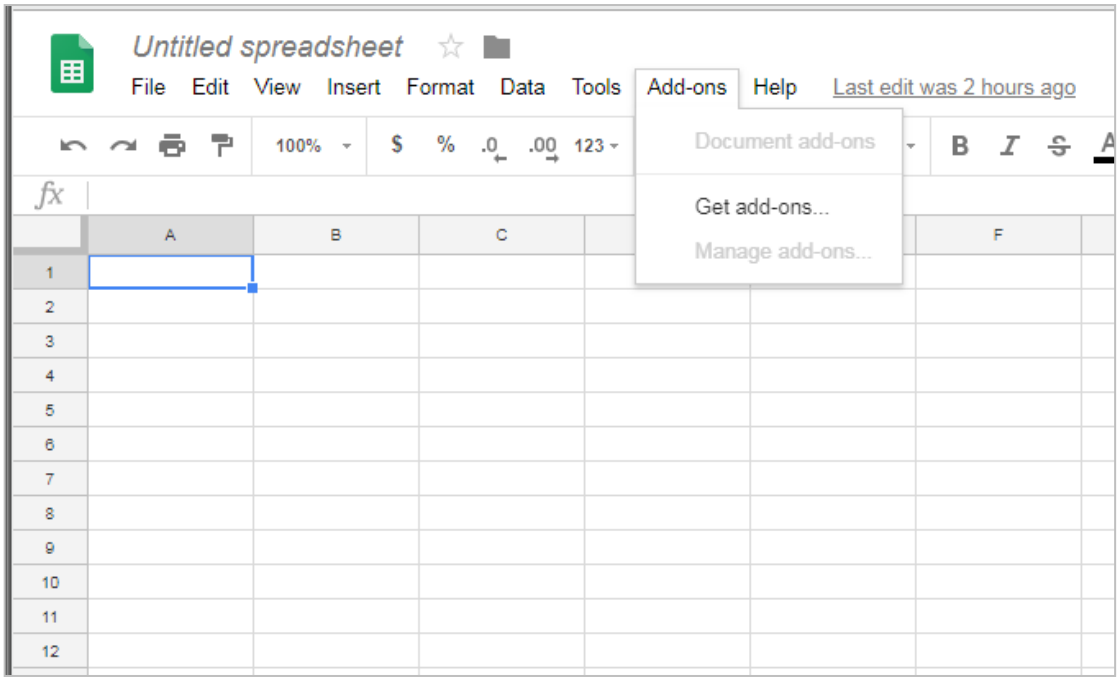
Using the ActivityEvents api, there is now a Google Sheets Add-on. This allows a user to install the add-on into their chrome account and when filling in a google sheet, to send data to the activity events api. This can then issue badges if interacting with appropriate rules.

Each row of the sheet is a 'rule activity', meaning that when the rules processor receives the activities from the sheet it treats each activity as a single entity, checking if the rule has been satisfied as it goes along.

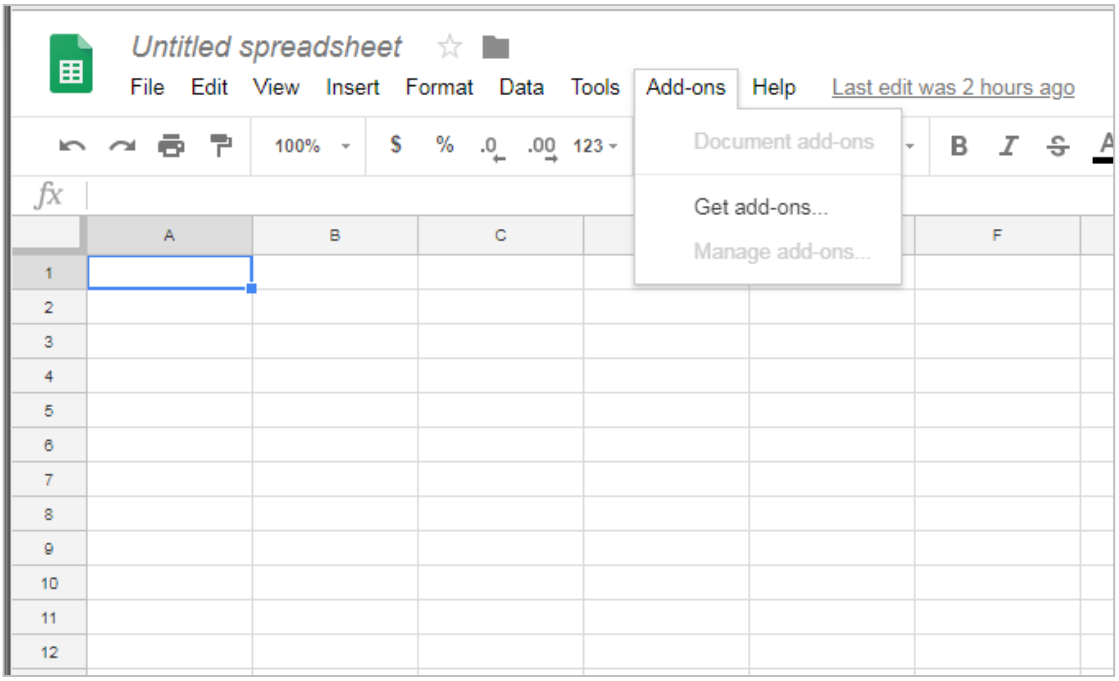
Walkthroughs

Google sheets add-on

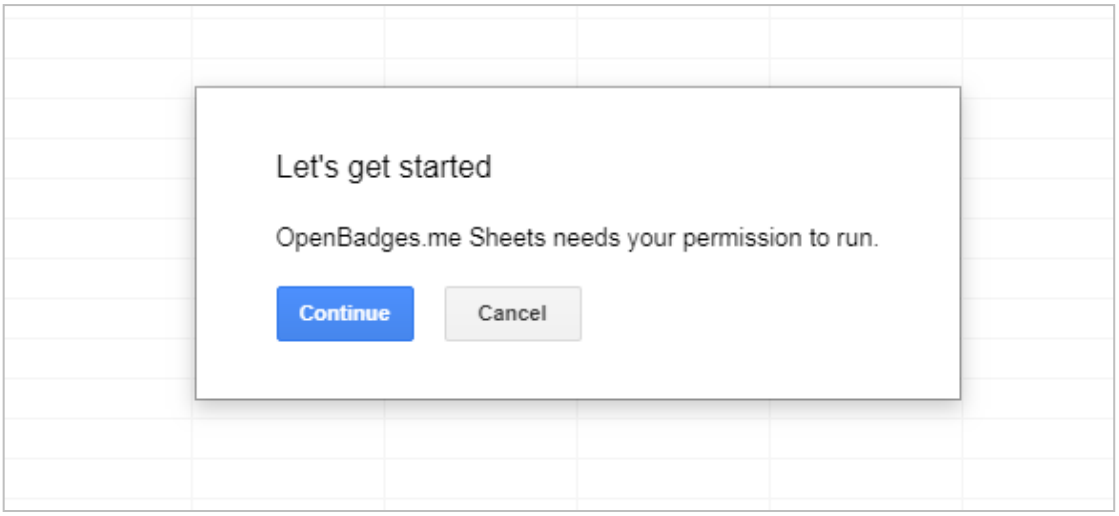
1. Sign into google sheets



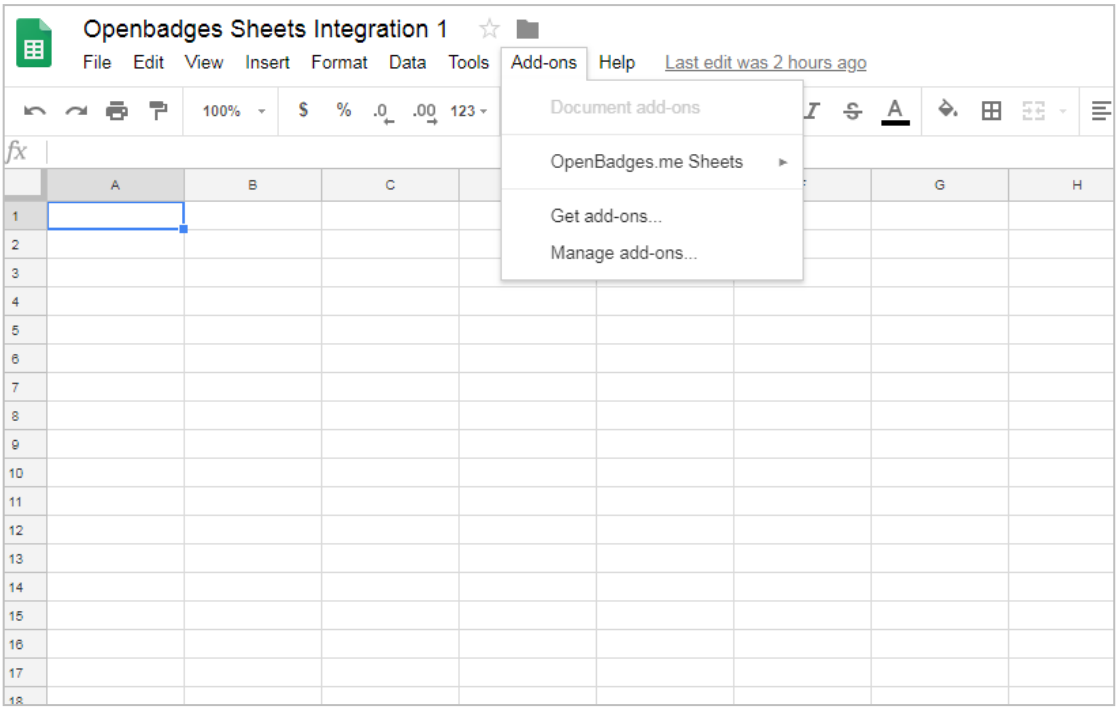
2. Click Add-ons > Get add-ons – note if the add-on has not been published yet, install using this private location https://chrome.google.com/webstore/detail/ohjonalgodncgealfnkhjmlphjepcjcc
3. Find the openbadges.me sheets add-on



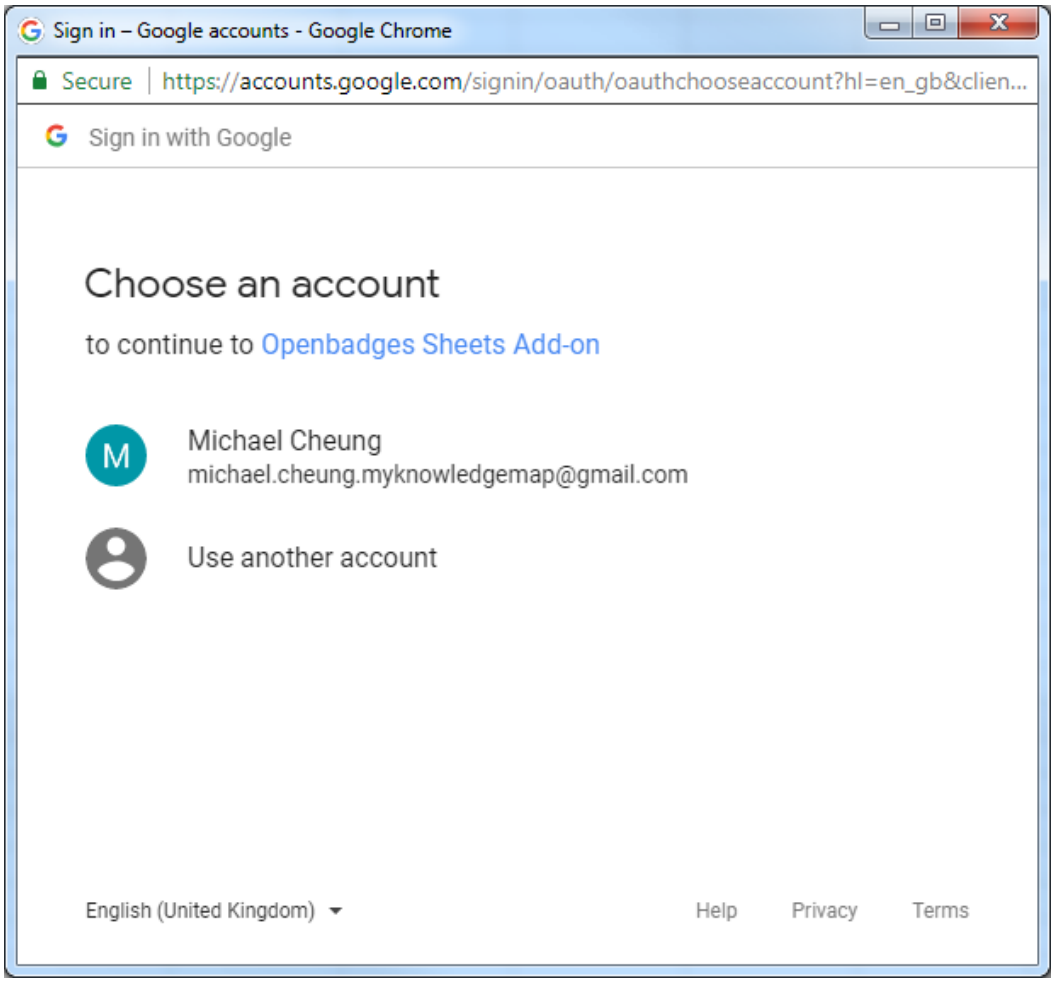
4. Give the add-on permission to run



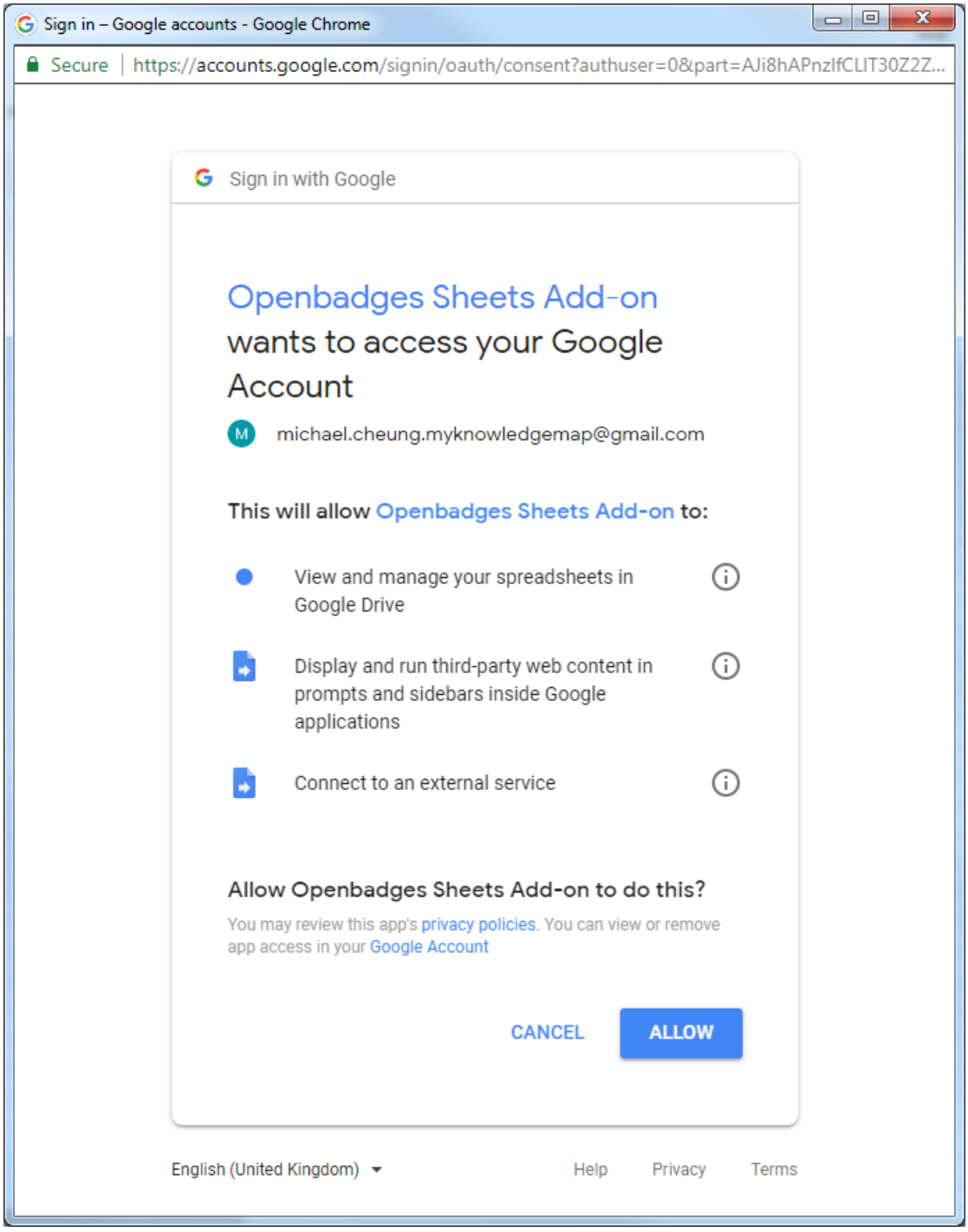
5. When it's installed you will see a new sheet and you can go to add-ons > Openbadges.me Sheets



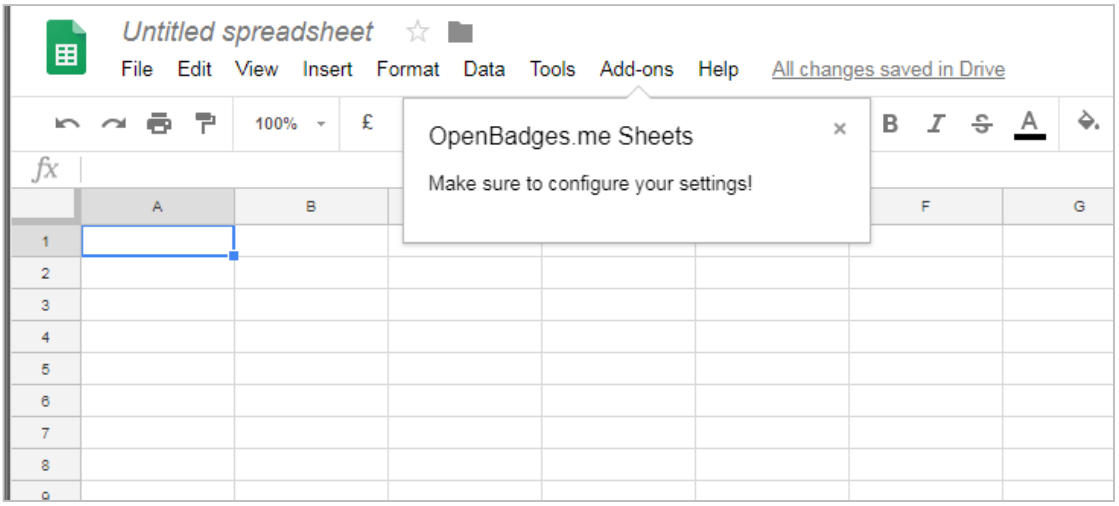
6. When prompted to continue to the add-on



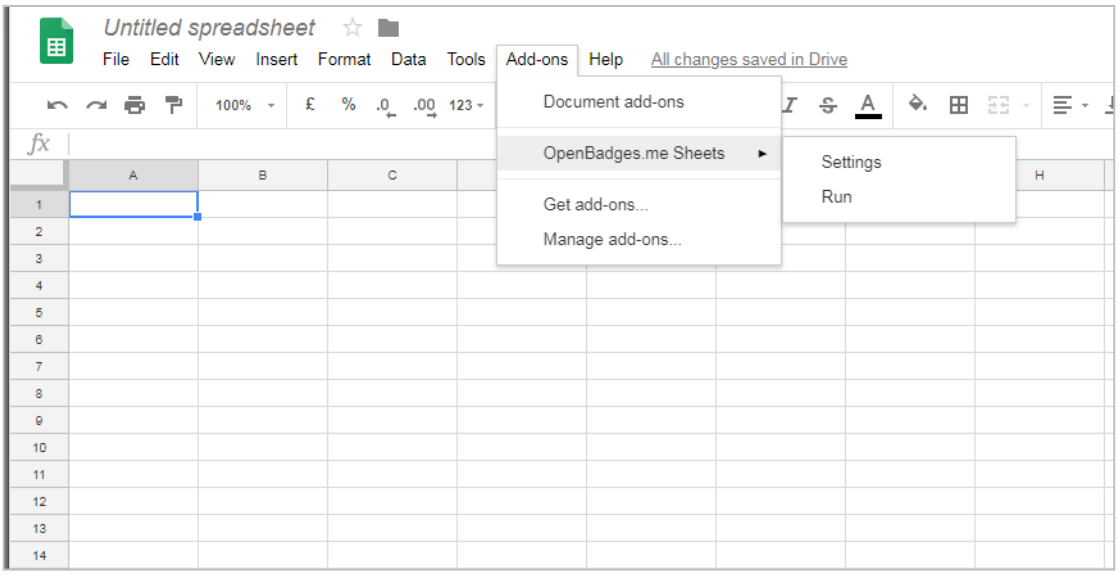
7. And allow it to have access to your google account



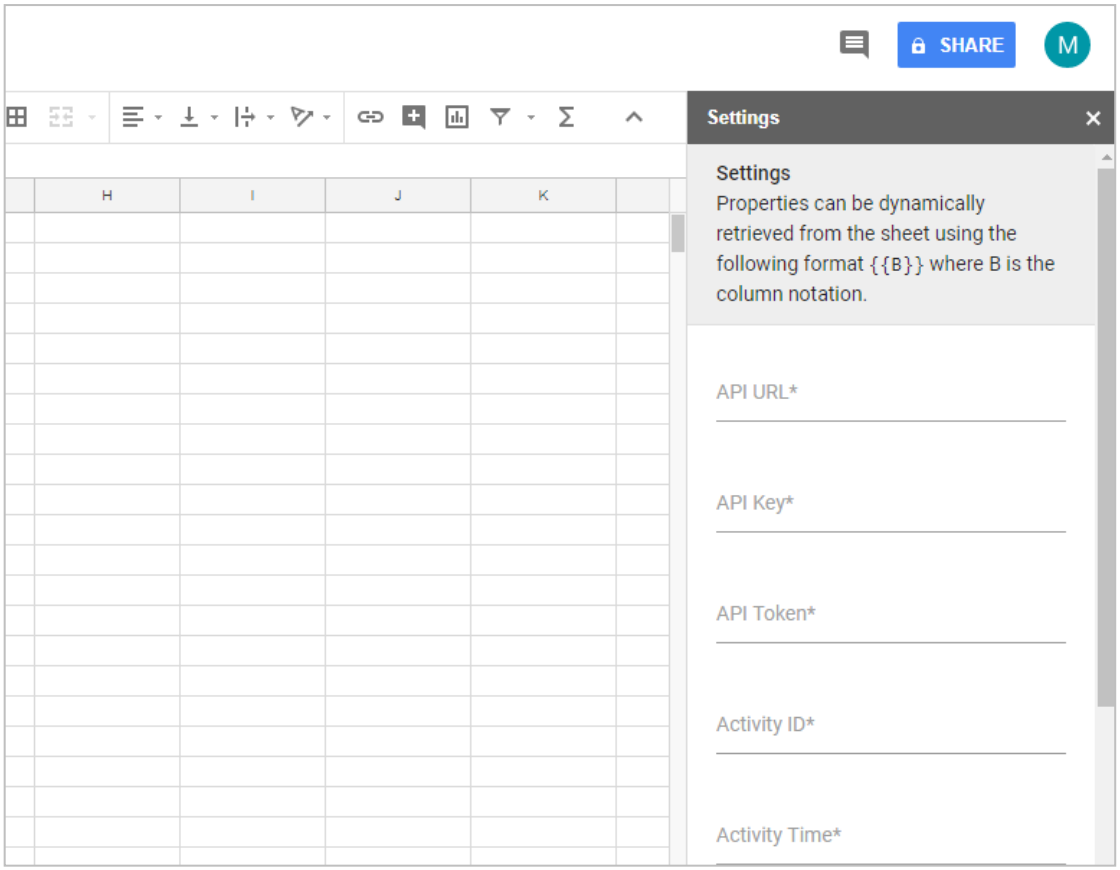
8. You will be notified to 'configure your settings'



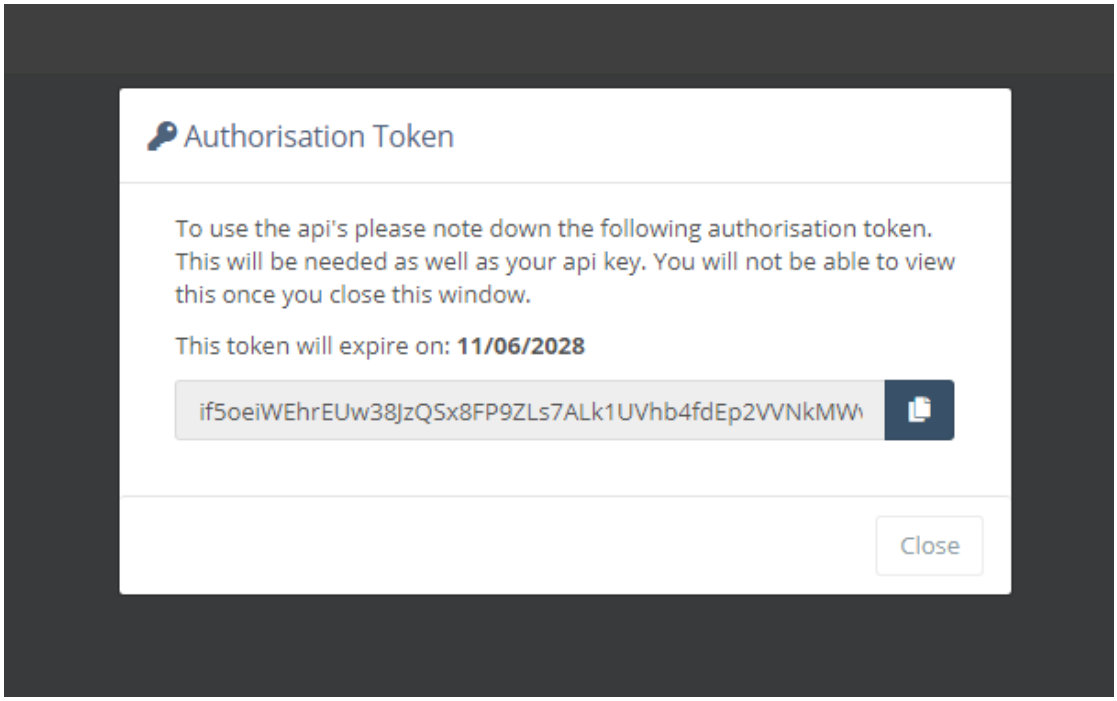
9. Go to add-ons > openbadges.me sheets > settings to open the settings drawer



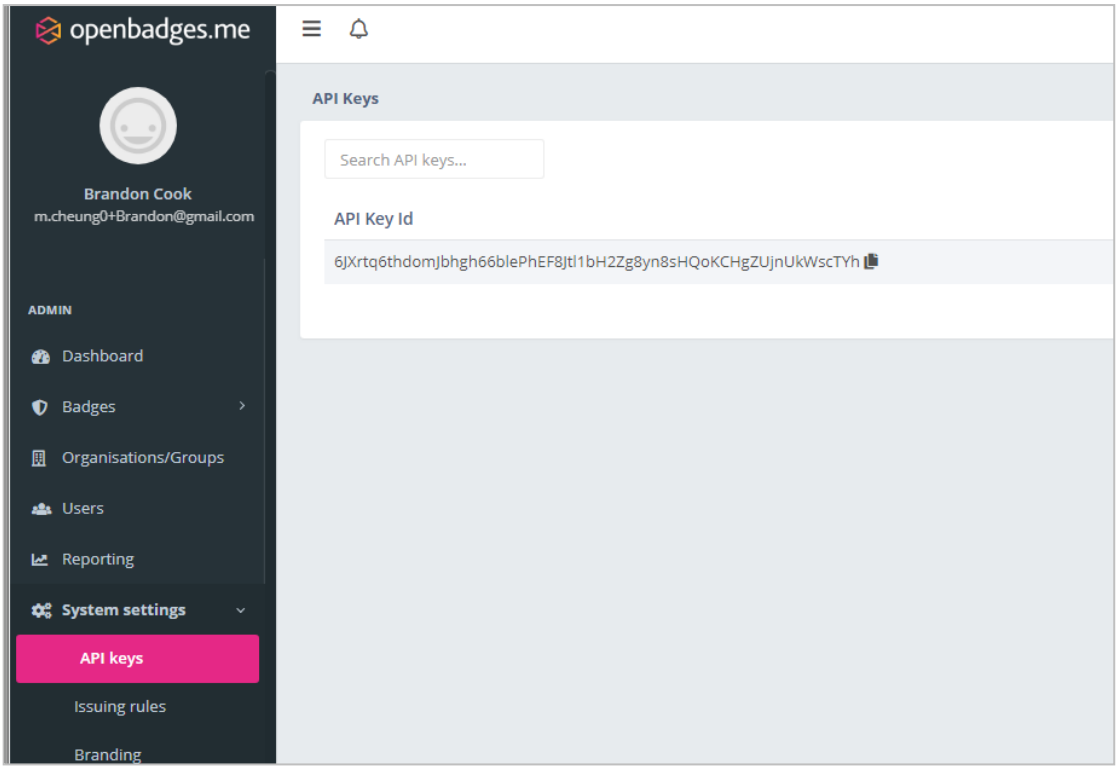
10. In the settings you will configure the openbadges api and authentication key and token to use



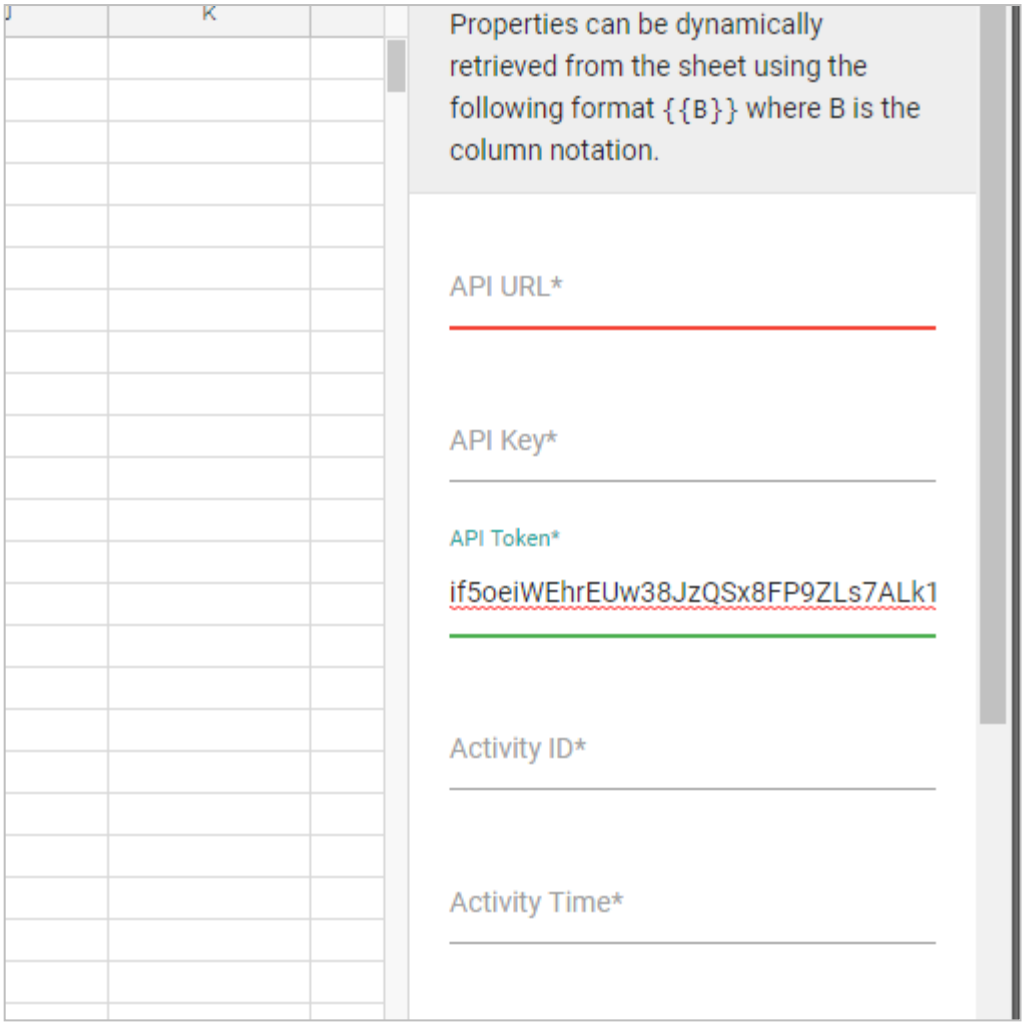
11. First sign into openbadges as an admin then go to System Settings > API keys > Create



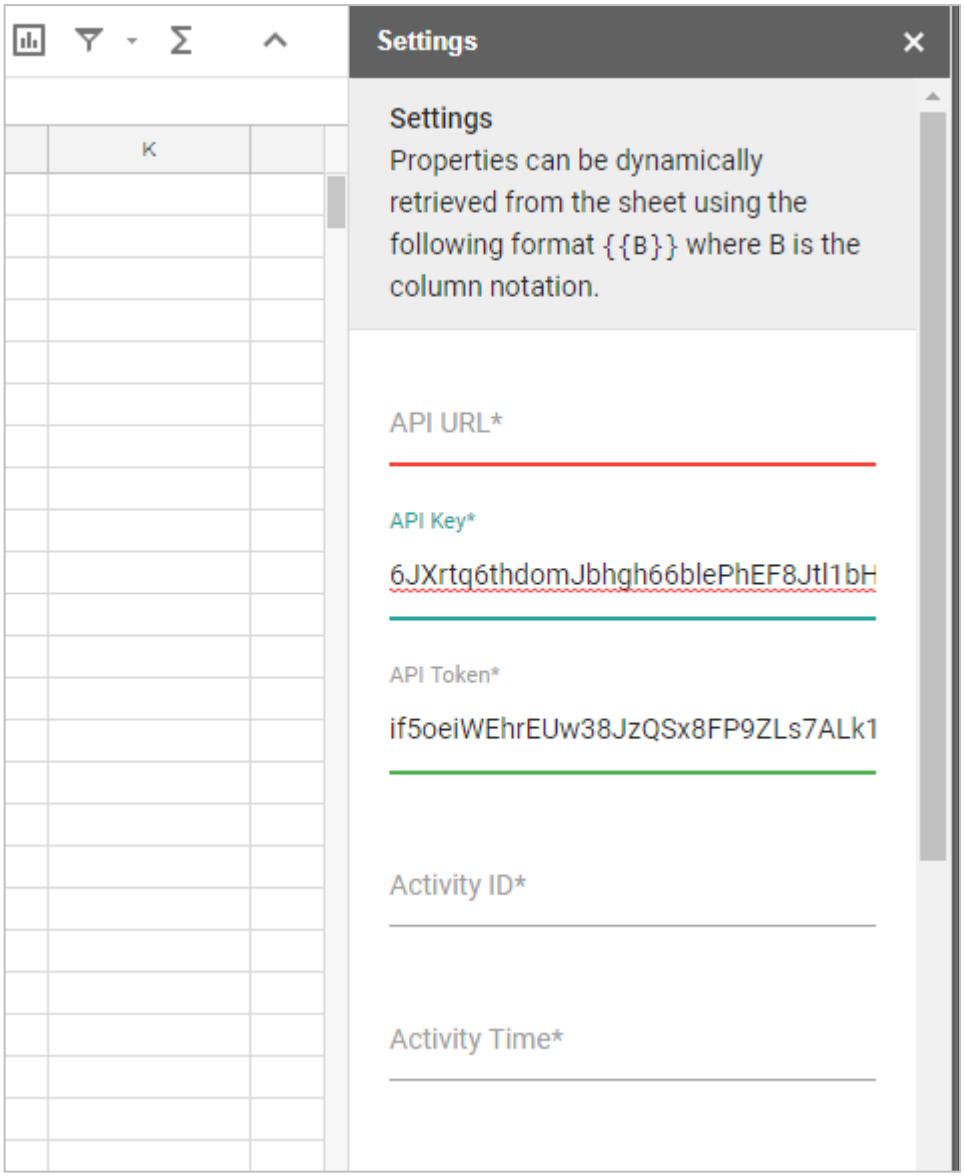
12. Note down the Authorisation Token as this will not be accessible once the modal has been closed
13. And once the modal is closed, note down the API Key Id



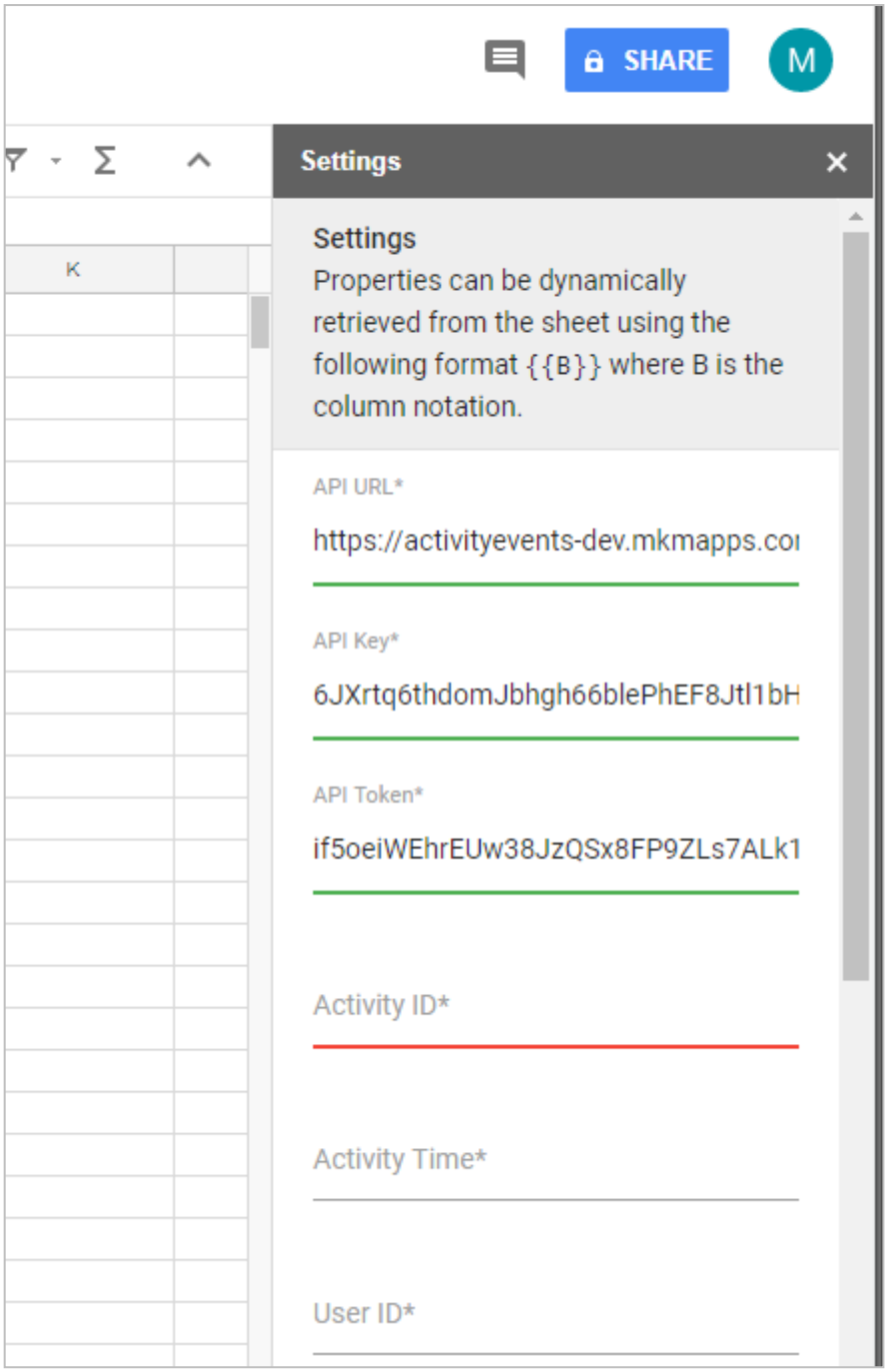
14. Now return to the google sheet and enter the api token



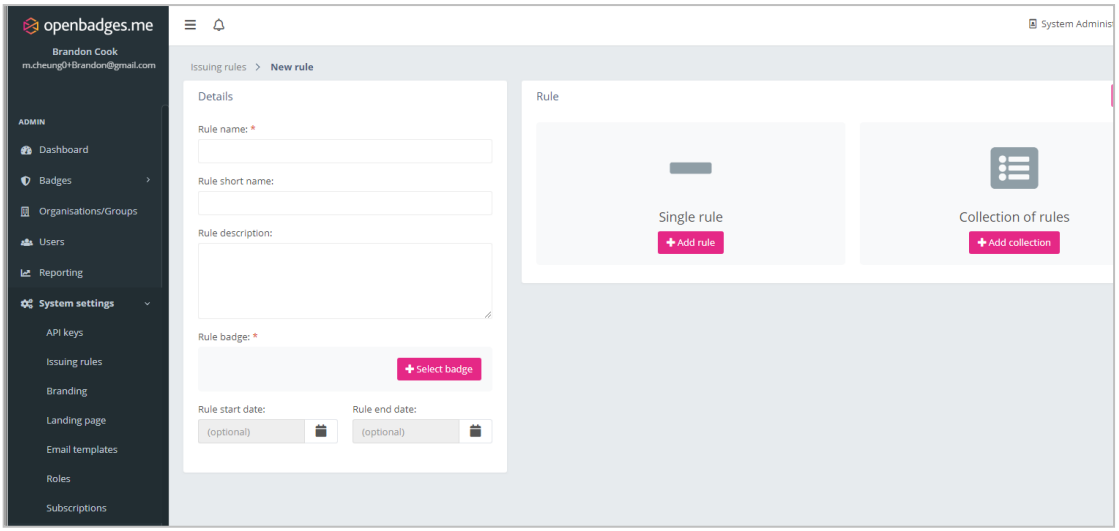
15. The api key



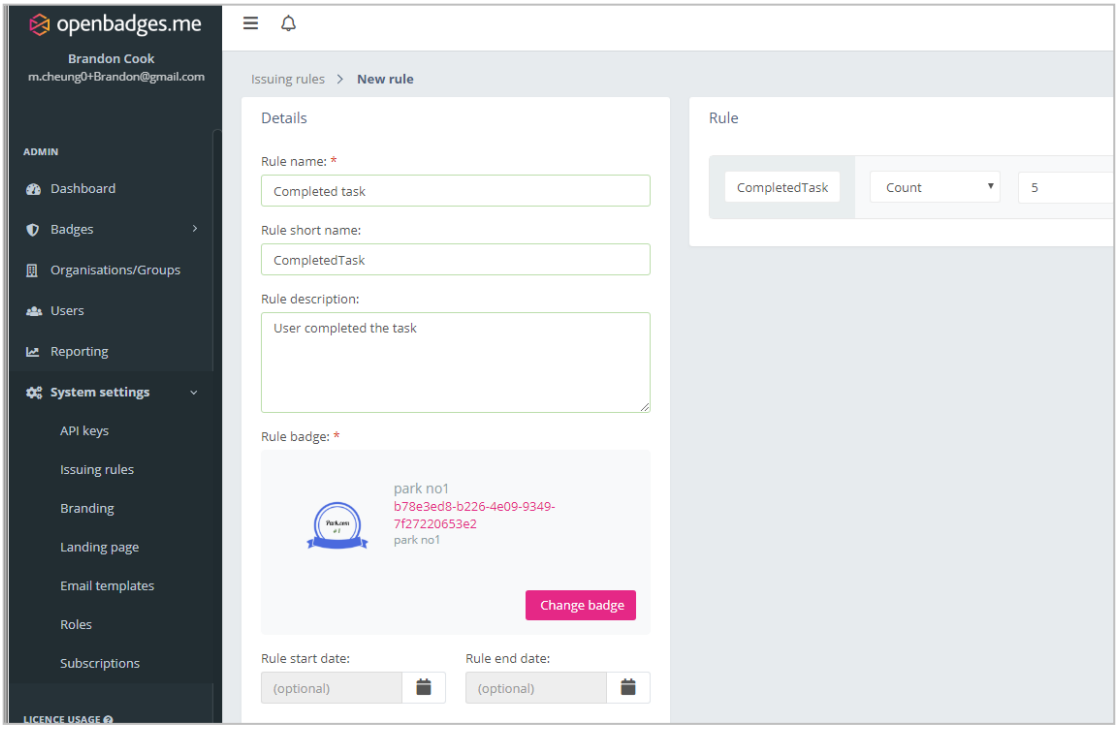
16. And enter the api url of 'https://activityevents-test.mkmapps.com/api/activityevents/bulk'



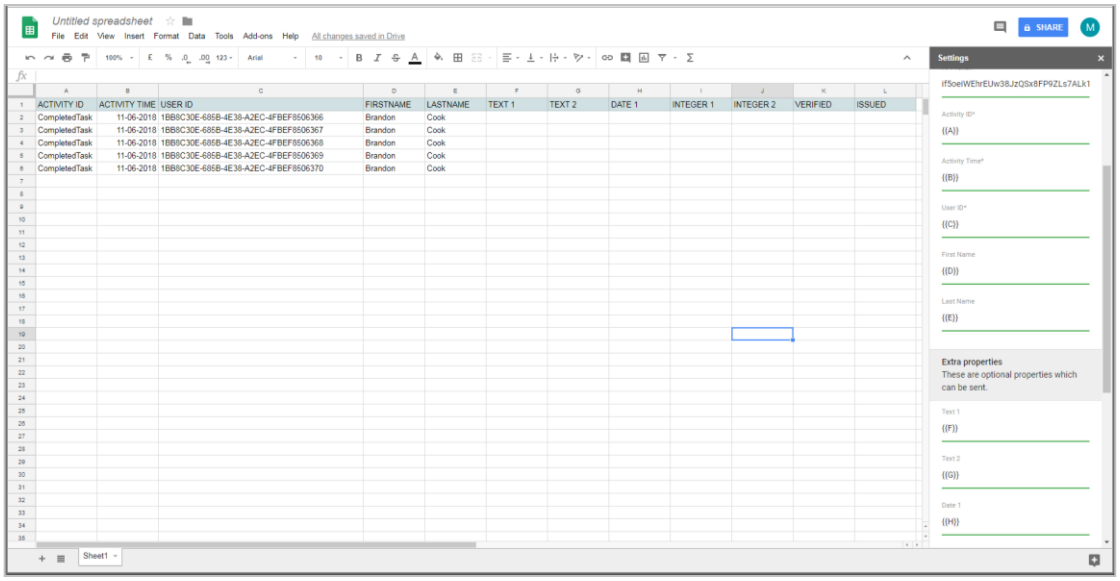
17. Finally, click ‘Save’ to save the settings
18. Now in openbadges go to System Settings > Issuing Rules > Create New Rule



19. Add a ‘Single rule’. In this example we’ve said the rule named ‘CompletedTask’ will be triggered when a user is submitted with 5 activities referencing the rule (referred to as activityid in the sheet add-on)



20. Back in the sheet, map columns to the settings fields using the {{COLUMN LETTER}} notation – note, the top row is reserved for headers and won’t be processed. In this example we have input 5 rows with the activityid ‘CompletedTask’ so that when run the ‘CompletedTask’ rule will be run for the user



21. Finally, click add-ons > openbadges.me sheets > run to run the sheet processor and issue the badge that was set up in the rule

Google Forms add-on

https://chrome.google.com/webstore/detail/openbadgesme-forms/hokfhhlaabikmmfpihalkjnclfcogiek

Activity fields

In order for a rule activity to be valid, the following fields must be set, per activity row

- ActivityId: the id for the 'Rule'
- Activity Time: the date the activity took place for the user
- User Id: the identifier for the user – an Email Address
- Firstname/Lastname: the user's firstname and lastname

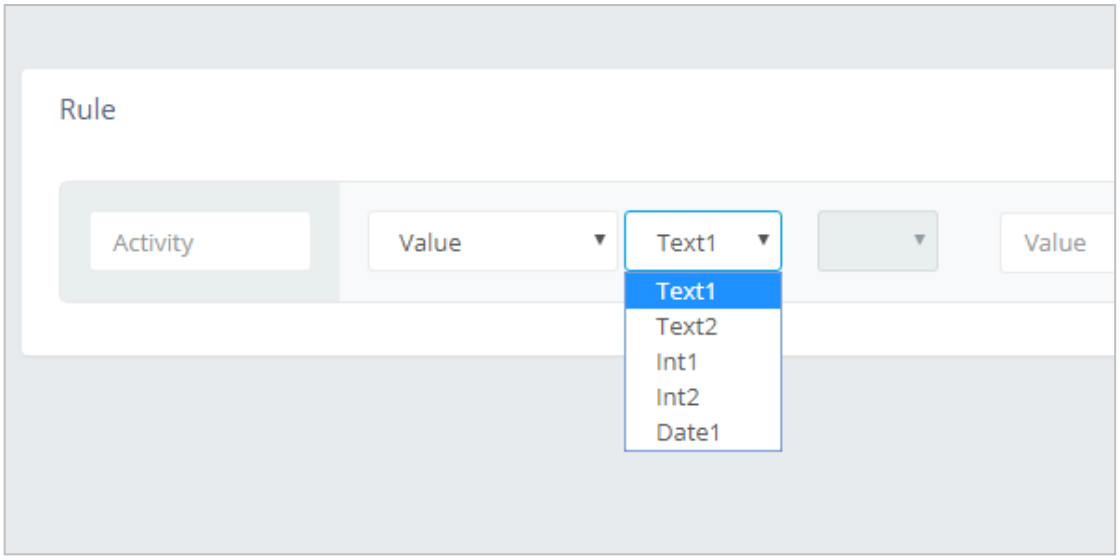
Rule Parameter Values and Match Operators

When setting up a rule, depending on the condition you set, parameter values are made available:

- Text 1
- Text 2
- Date 1
- Integer 1
- Integer 2

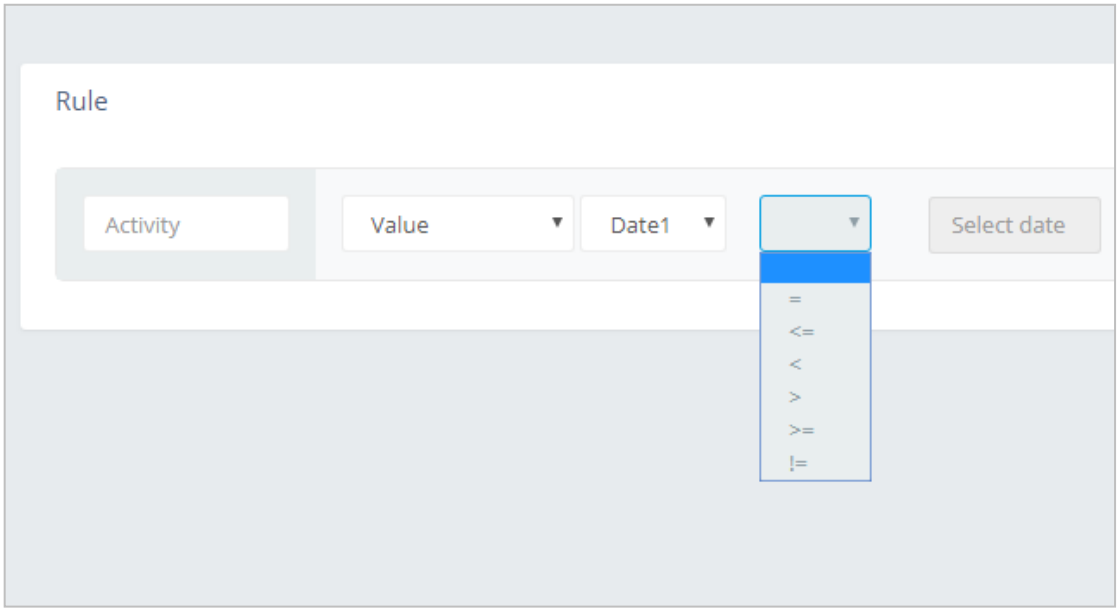
These allow a rule to have syntax such as

- 'if the value supplied is equal to the value of <Text 1>, then be triggered'



Likewise, value match operators are made available according to the selected condition:

- = 'equal to'
- <= 'less than or equal to'
- < 'less than'
- > 'more than'
- >= 'more than or equal to'
- != 'not equal to'
- - 'between'

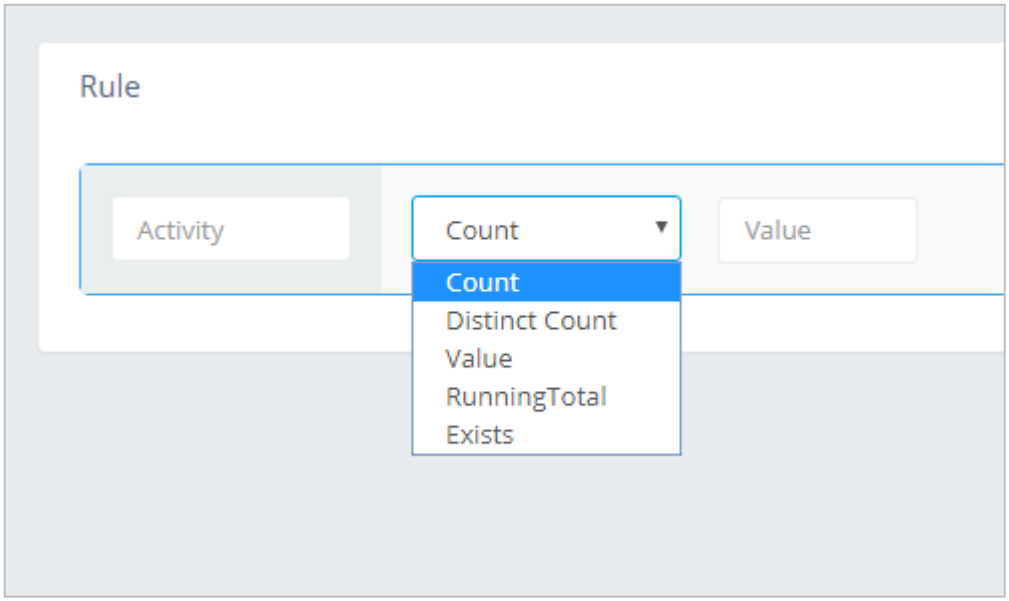


Rule Conditions

A rule must specify a condition to operate on:

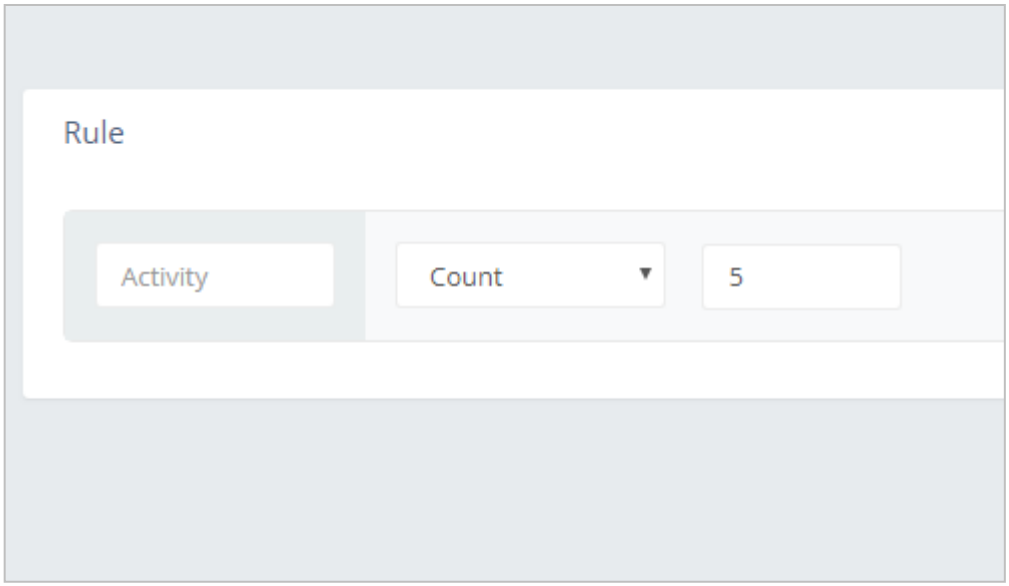
- Count
- Distinct Count
- Value

- Running Total
- Exists



Count

The count condition specifies that *at least* the given number of rule activities must be supplied by the spreadsheet for a unique user



Example

Here the user has 5 rule activities for the 'CompletedTask' rule which would satisfy the 'Count 5' condition described above.

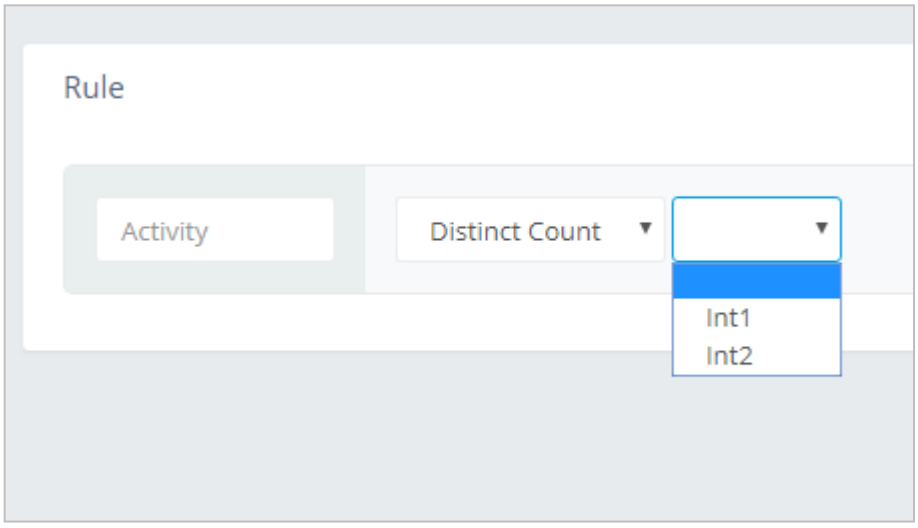
Untitled spreadsheet

File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive

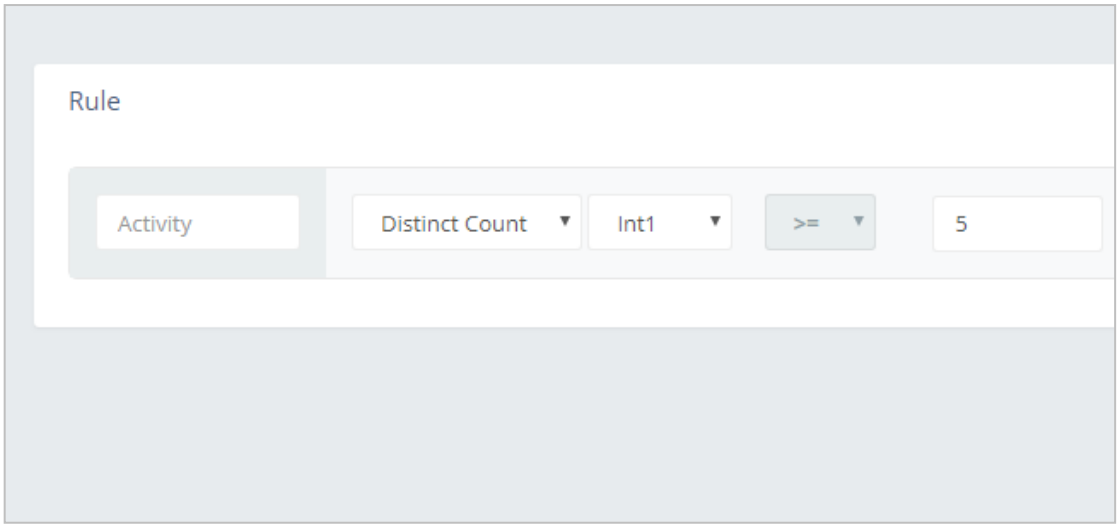
	A	B	C	D	E	
1	ACTIVITY ID	ACTIVITY TIME	USER ID	FIRSTNAME	LASTNAME	TEX
2	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook	
3	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506367	Brandon	Cook	
4	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506368	Brandon	Cook	
5	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506369	Brandon	Cook	
6	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506370	Brandon	Cook	
7						
8						
9						
10						

Distinct Count

Distinct count specifies that the condition matches on either Int 1 or Int 2.

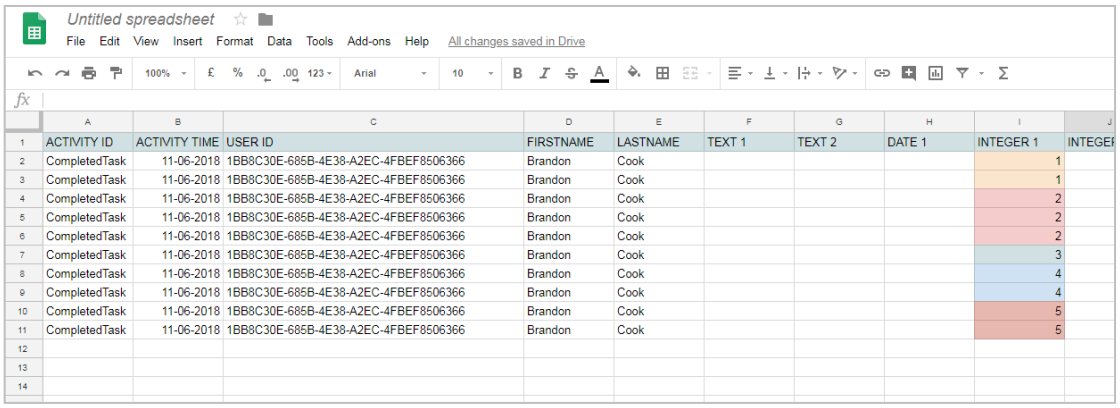


By specifying that either Int 1 or Int 2 have distinct values *more than or equal to* a given number of times, the rule is only satisfied if activities for a user are passed to it and the value in Int 1 or Int 2 are distinct more than or equal to the set number of times.



Example

Here, a user has 10 activities, and the Int 1 fields have some repeating values, however, on aggregate there are 5 distinct values, and so the rule described above will be satisfied:



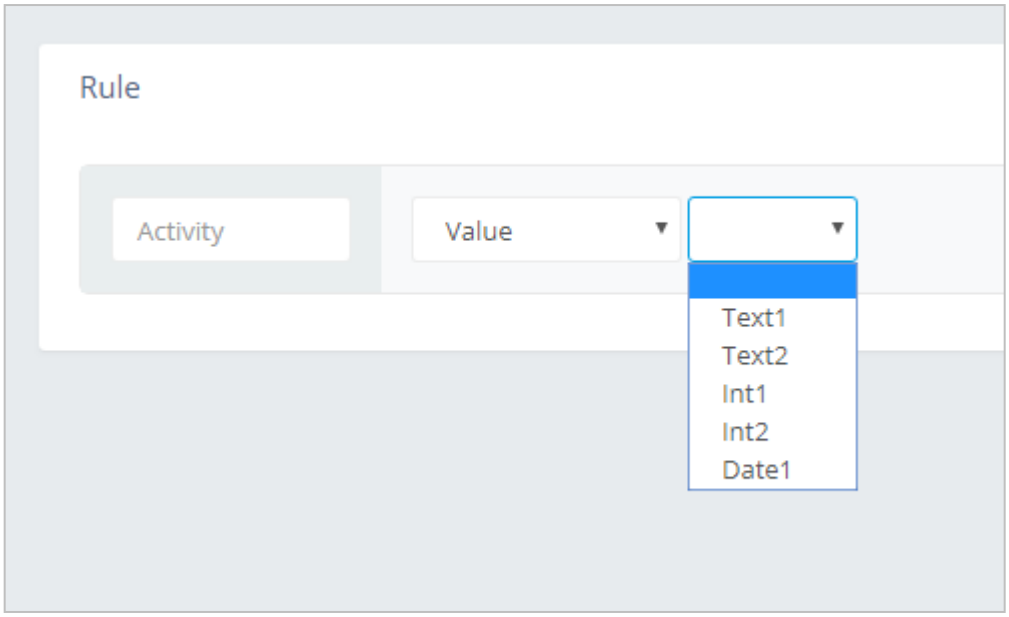
Untitled spreadsheet

File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive

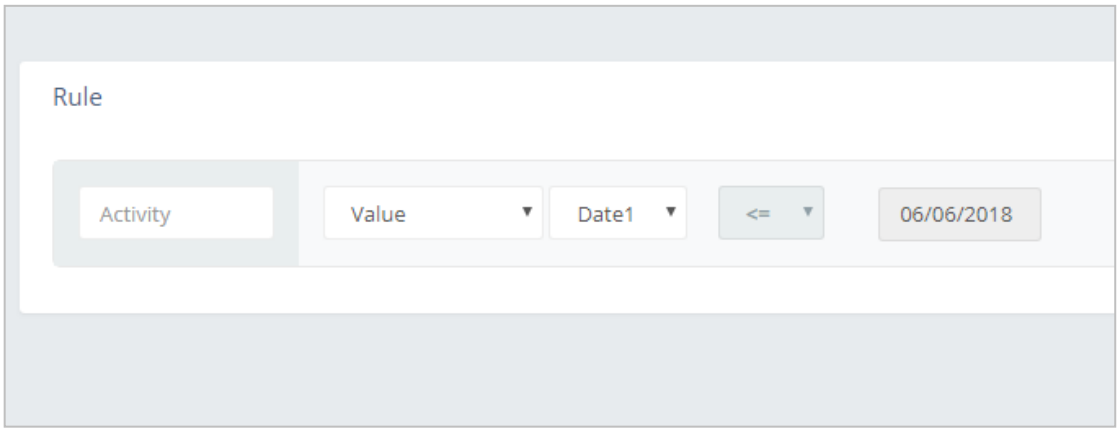
	A	B	C	D	E	F	G	H	I	J
1	ACTIVITY ID	ACTIVITY TIME	USER ID	FIRSTNAME	LASTNAME	TEXT 1	TEXT 2	DATE 1	INTEGER 1	INTEGE
2	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				1	
3	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				1	
4	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				2	
5	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				2	
6	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				2	
7	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				3	
8	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				4	
9	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				4	
10	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				5	
11	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				5	
12										
13										
14										

Value

A value condition specifies that the selected parameter must have a value that satisfies the operator and supplied value.

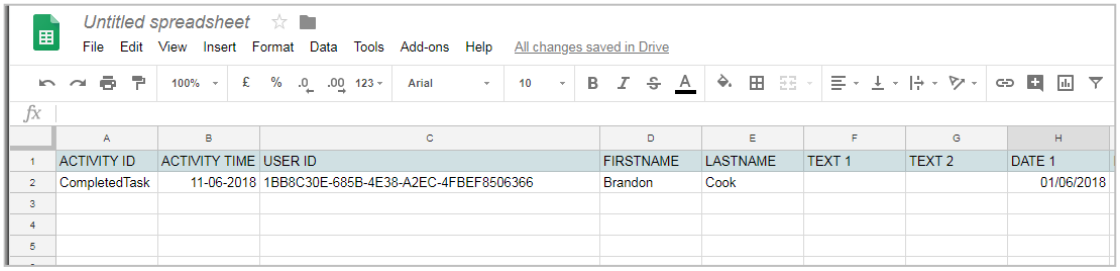


In this example the rule states that a given activity must have a value in Date 1 that is less than or equal to '06/06/2018'



Example

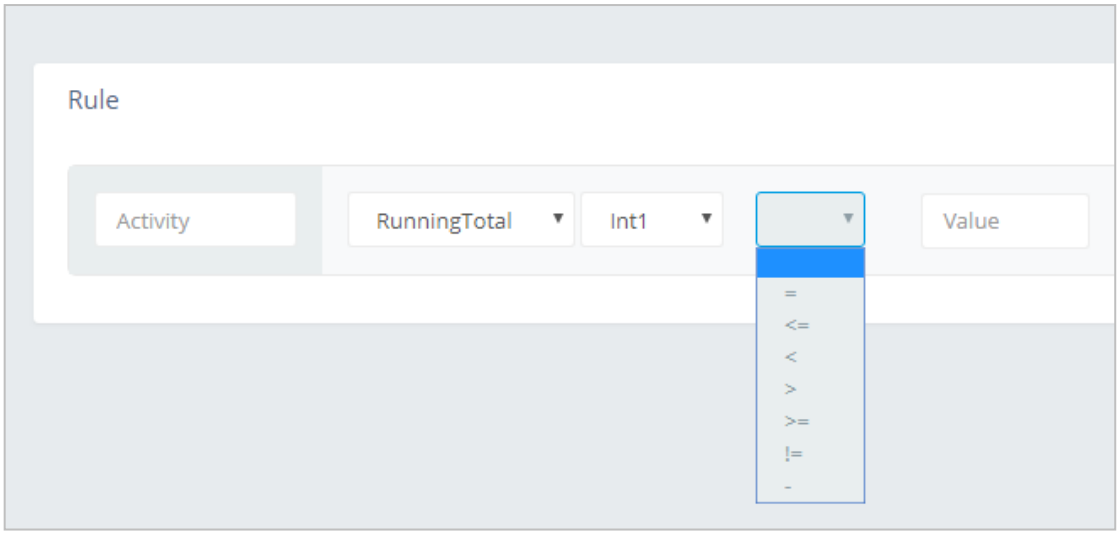
The user has an activity with a Date 1 value of 01/06/2018 and so this will satisfy the rule above:



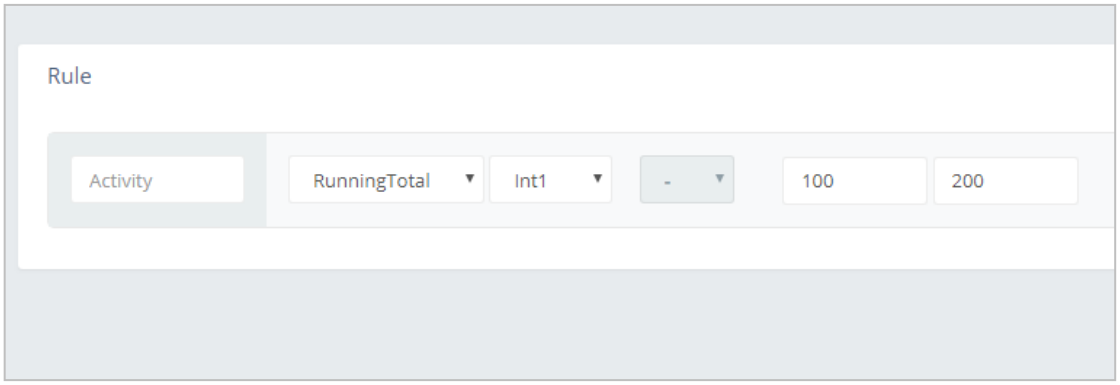
	A	B	C	D	E	F	G	H
1	ACTIVITY ID	ACTIVITY TIME	USER ID	FIRSTNAME	LASTNAME	TEXT 1	TEXT 2	DATE 1
2	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook			01/06/2018
3								
4								
5								

Running Total

For running total a rule expects the total values in either Int 1 or Int 2 to be match the selected operator and value

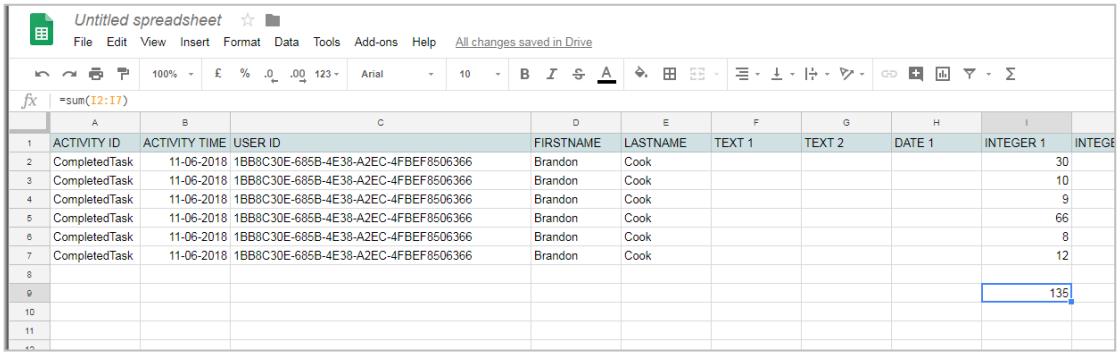


For example here the rule expects a running total of between 100 and 200 in the total of Int 1 values:



Example

The user has 6 activities with different values in Int 1, and the running total is 135, so this satisfies the rule:



Untitled spreadsheet

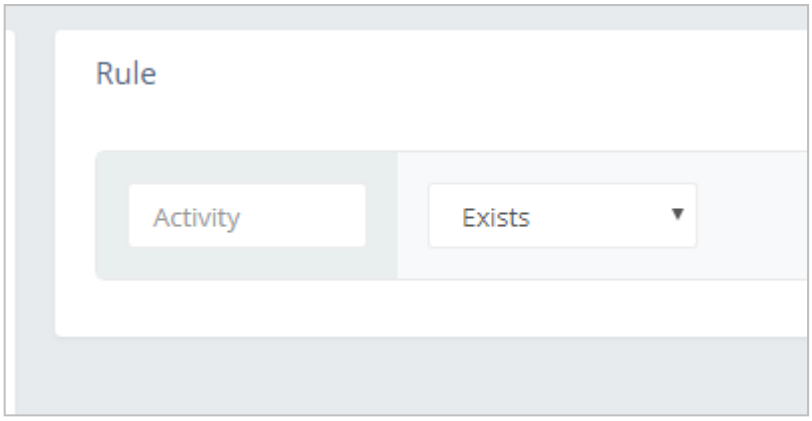
File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive

=sum(I2:I7)

	A	B	C	D	E	F	G	H	I	
1	ACTIVITY ID	ACTIVITY TIME	USER ID	FIRSTNAME	LASTNAME	TEXT 1	TEXT 2	DATE 1	INTEGER 1	INTEGE
2	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				30	
3	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				10	
4	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				9	
5	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				66	
6	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				8	
7	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook				12	
8										
9									135	
10										
11										

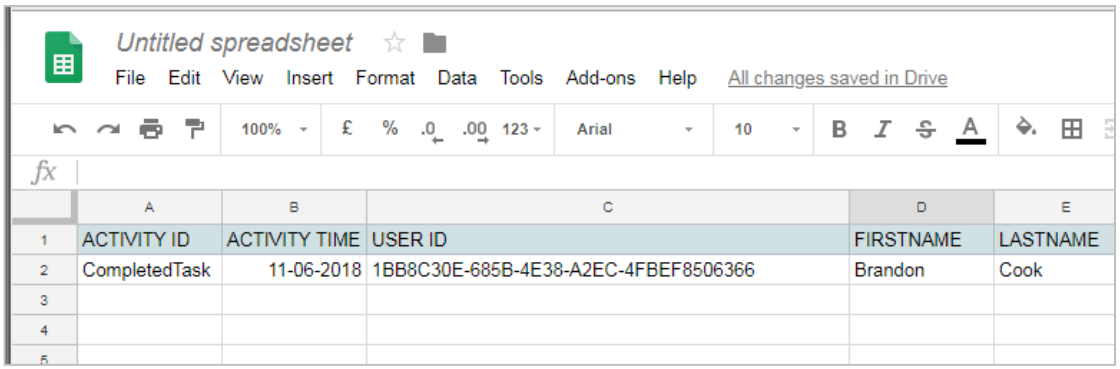
Exists

The exists condition simply expects the rule activity to exist for a user.



Example

The user has one activity for the rule, so this satisfies the 'exists' condition:

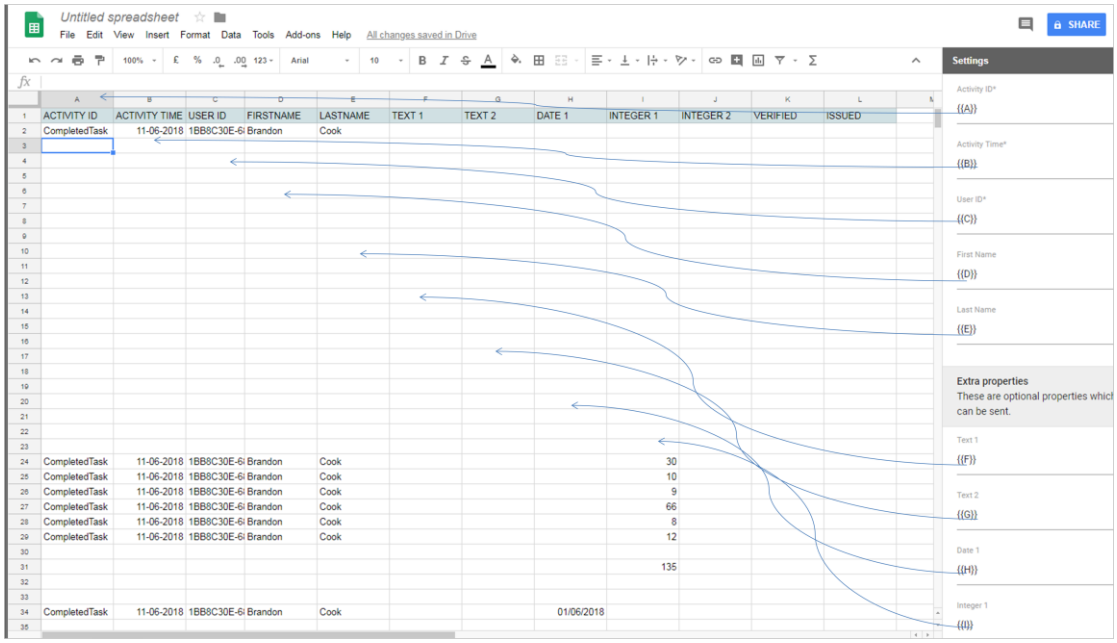


	A	B	C	D	E
1	ACTIVITY ID	ACTIVITY TIME	USER ID	FIRSTNAME	LASTNAME
2	CompletedTask	11-06-2018	1BB8C30E-685B-4E38-A2EC-4FBEF8506366	Brandon	Cook
3					
4					
5					

Field mapping

The settings fields use a {{COLUMN LETTER}} notation meaning that you can customise what order you want the columns to be in when sent back to the activity events api. In this example {{A}} is Activity Id, {{B}} is Activity time, {{C}} is User Id and so on. This means that different activities for different users with different values can be sent in one go by adding the relevant data in each row:

Note: the top row is reserved for headers and can be used for labeling purposed.



Enhancements and Bug Fixes

[Report changes in this release. Include the issue/ticket number, type and one-line summary. Include issues that were highlighted in the "What's New" section above.]

Known Problems and Workarounds

[State all known defects discovered in this release or in previous releases that are still not resolved. Include information on workarounds from the issues.]

Supporting Documentation

[Attach any supporting documentation such as the RFQ, Functional Specification, design documents, etc. Please ensure these are clearly annotated with any additions, omissions, or alterations.]

Browser Compatibility

[List the browsers this development is required to be compatible with.]

Have you tested all the above browsers and can confirm that the new features/fixes/enhancements display correctly?

YES/NO (Delete as appropriate.)

If NO, please provide the reason.

Required Software Downloads and Recommended Versions

[Enter the software downloads and versions that are required for this development, eg. Java, Flash player.]

QA Checklist

Below is basic testing and quality assurance checklist. Please confirm that all of the following have been applied/checked before releasing to testing.

1. **Forms and Search Fields**
    a) Max character lengths
    b) Non-alphanumeric characters, numbers and letters, html tags, eg. !"£$%^&*()_+=-{}:@~<>?[];'#,./\|`¬ <text> <text< abc 123
    c) Validation, eg. times, dates, numbers, numbers only etc.
    d) Required fields
    e) Red error messages

2. **Usability, Accessibility and Navigation**
    a) Breadcrumbs – are correct and hyperlinked appropriately
    b) Correct browser tab names
    c) Tool tips/Alt tags – check they exist and are correct
    d) Success/Failure notifiers, eg. When adding/editing/deleting/uploading/saving
    e) Confirmation dialog box – asking user to approve requested operation, eg. to delete something
    f) Simple and logical navigation (eg. Back, Return to... or cancel links)
    g) Correct mouse cursors

3. **Content – Quality and Consistency**

    a) Correct and user friendly instructional text

    b) Consistency of buttons, titles, headers etc. (eg. Title case, sentence case, camel case, size, font, colour)

    c) Spelling

    d) Grammar

    e) Punctuation

Have all the above applicable items been tested prior to release to testing?
YES/NO (Delete as appropriate.)

Reviews and Demos

[Please enter the dates that reviews and demos have taken place or are scheduled to occur and confirm that a smoke/sanity test has been conducted.]

Date peer review conducted:
Name of colleague who conducted the review:

Date of developer to tester demo/walkthrough:

Has a smoke test been conducted on the test environment to qualify the build?
YES/NO (Delete as appropriate)

Installation and Upgrade Notes

[List any specific database updates, web.config settings or any other relevant information required for installation/upgrade.]

Database Updates

Web.Config Settings

Other

The edit page for the google **forms** add-on:
https://chrome.google.com/webstore/developer/edit/hokfhhlaabikmmfpihalkjnclfcogiek

The edit page for the google **sheets** add-on:
https://chrome.google.com/webstore/developer/edit/ohjonalgodncgealfnkhjmlphjepcjcc

Dependencies

[List any dependencies.]

Environments

[Enter the development, testing and live environments. List URLs and also log ins if appropriate.]

Development

Mobile Platform

URL

Database

User Login Data

Testing

Mobile Platform

URL

Database

User Login Data

Demo

Mobile Platform

URL

Database

User Login Data

***Please upload completed release notes to the appropriate folder in SharePoint**