

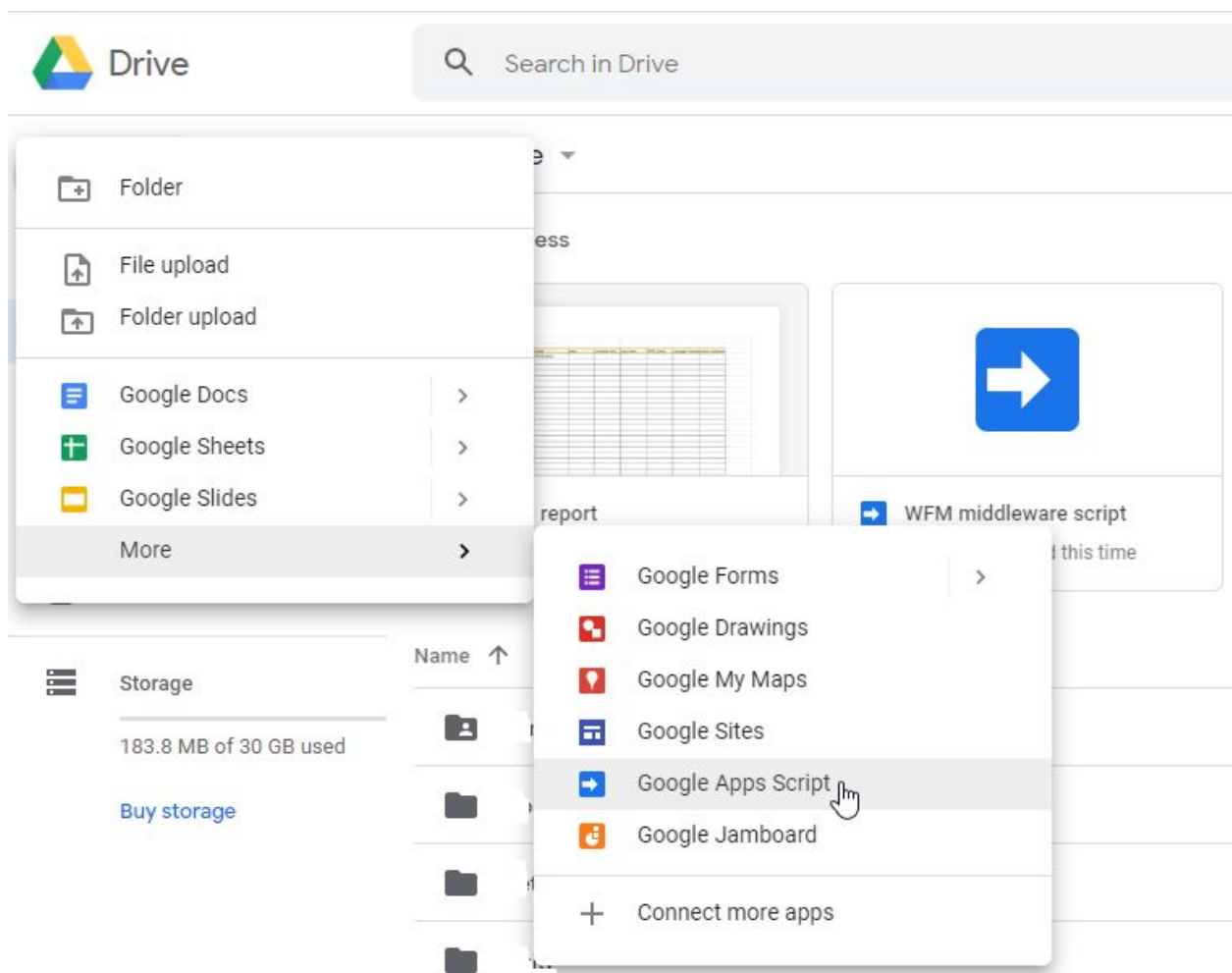
Init source in google drive

1. Access <https://drive.google.com/> and create new folder WFH

2. In folder WFH, we will create new file Google Apps Script

If you don't see "Google Apps Script",

please click New --> More --> "Connect more apps", and then search with keyword "Google Apps Script".

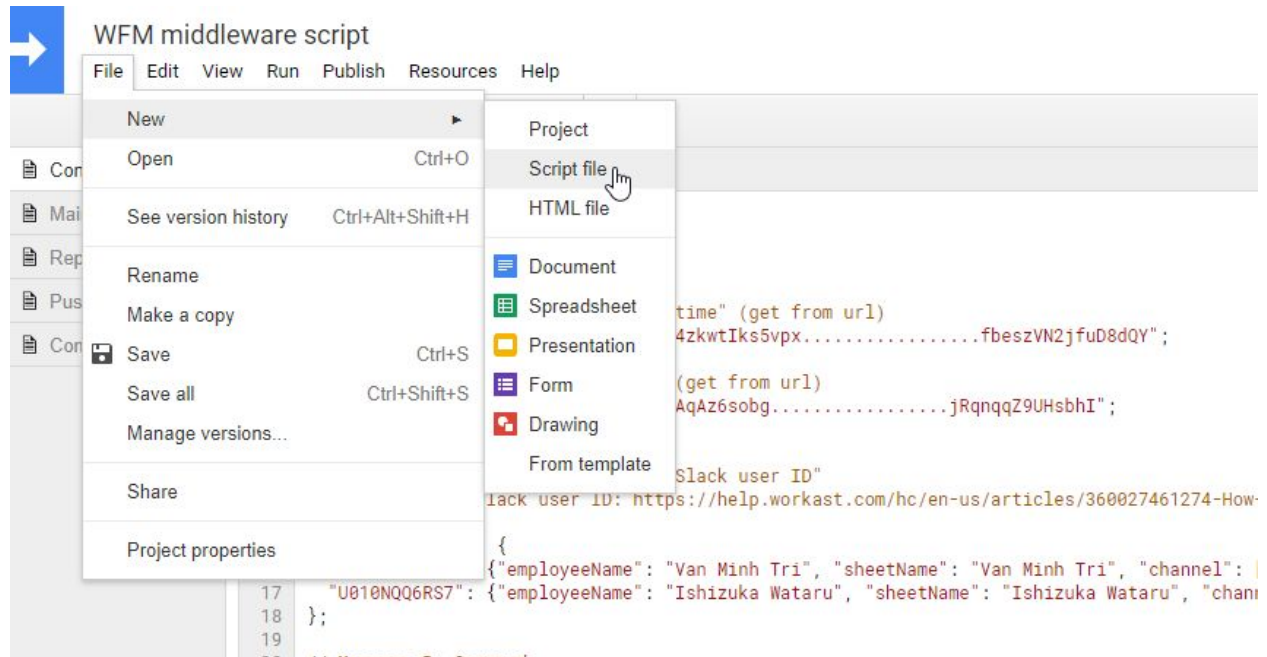


a) Double click in new file, google will bring you to Apps Script Editor

b) Create new 5 files in Apps Script Editor

Content of file, please get it in here:

<https://github.com/grevo-vn/Work-From-Home/tree/master/GoogleAppsScript>

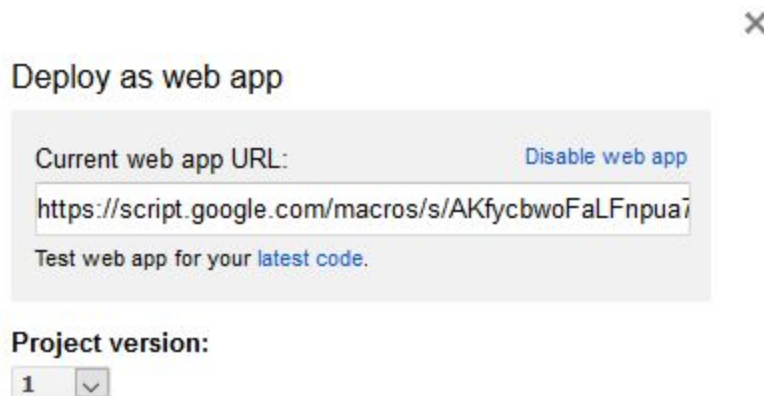


c) Deploy script as web app

Choose Publish --> Deploy as web app...

Set Who has access to the app = "Anyone, even anonymous"

And then click Deploy.



Copy "Current web app URL", we will use it for "Slack Apps".

Question: Why do we need a Google Apps Script file?

Answer: We need it as a web server that will receive response from slack, save data to google sheet and post message to some groups in slack.

PS: If you want to distribute or update script, please see more information of Google Apps Script in <https://developers.google.com/apps-script>

We will come back with "Google Apps Script file" again to update the configuration after we set up slack apps.

Set up Slack Apps

1. Create new slack apps

Go to https://api.slack.com/apps?new_app=1

Enter your "App Name", choose "Slack Workspace" and then click button "Create App"

Create a Slack App



App Name

e.g. Super Service

Don't worry; you'll be able to change this later.

Development Slack Workspace

Development Slack Workspace



Your app belongs to this workspace—leaving this workspace will remove your ability to manage this app. Unfortunately, this can't be changed later.

By creating a Web API Application, you agree to the [Slack API Terms of Service](#).

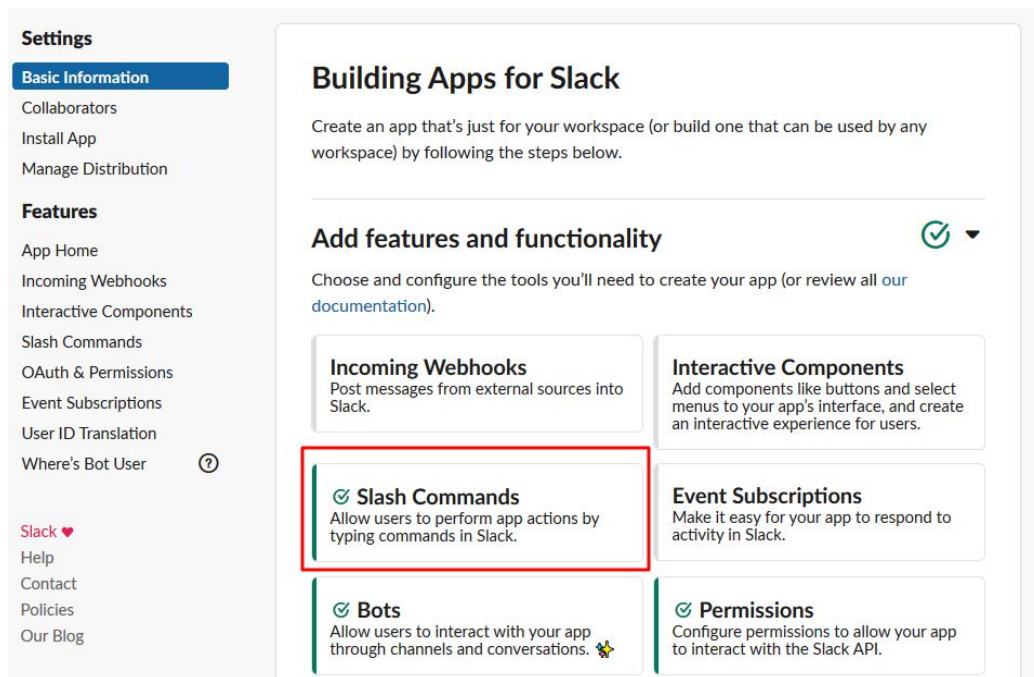
Cancel

Create App

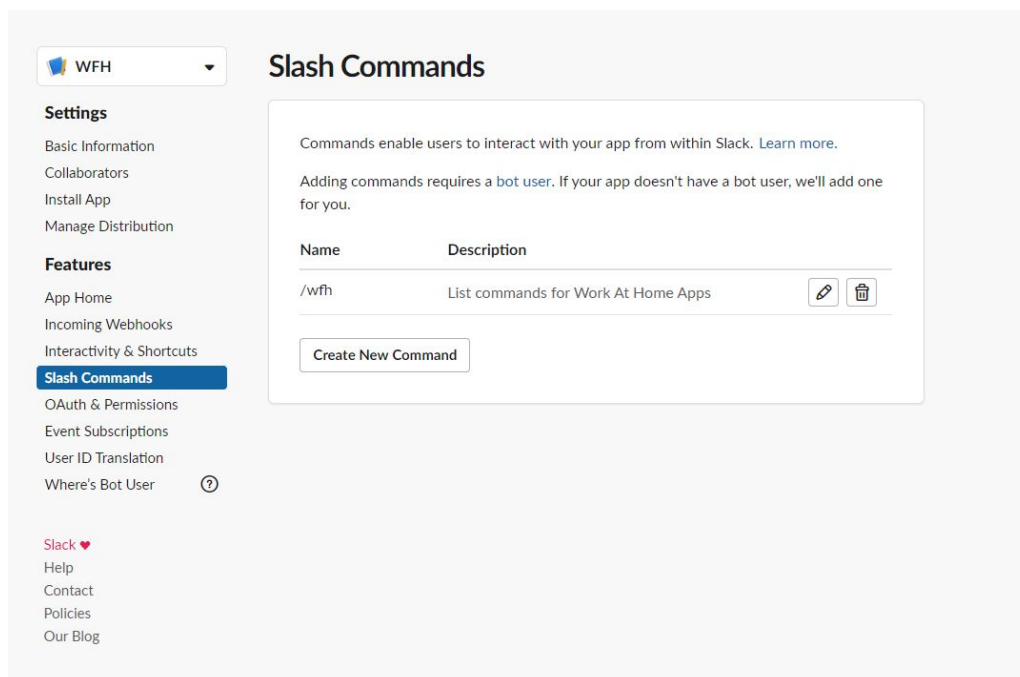
P/S: More information about Slack Apps, you can find out in here <https://api.slack.com/start/overview>

2. Set up Slash Command

a) Click on "Slash Commands"



b) Click button "Create New Command"



c) Setting command

Command	<input type="text" value="/wfh"/> ⓘ
Request URL	<input type="text" value="https://script.google.com/macros/s/..."/> ⓘ
Short Description	<input type="text" value="List commands for Work At Home Apps"/>
Usage Hint	<input type="text" value="start -msg [string] rest -msg [string] re..."/> <small>Optionally list any parameters that can be passed.</small>
Escape channels, users, and links sent to your app <input type="checkbox"/>	
Unescaped: @user #general	
Preview of Autocomplete Entry	
<div><div>Commands matching "wfh"</div><div><div>WFH</div><div>/wfh start -msg [string] rest -... List commands for Work At ...</div></div><div><div>+</div><div>/wfh</div></div></div>	

Command: Enter “/wfh”, this is what you need for running WFH APP on Slack.

Request URL: Enter "Current web app URL" that we get from [Google Apps Script](#).

Short Description: Shows what kind of app this is. Should be easy to understand.

Usage Hint:

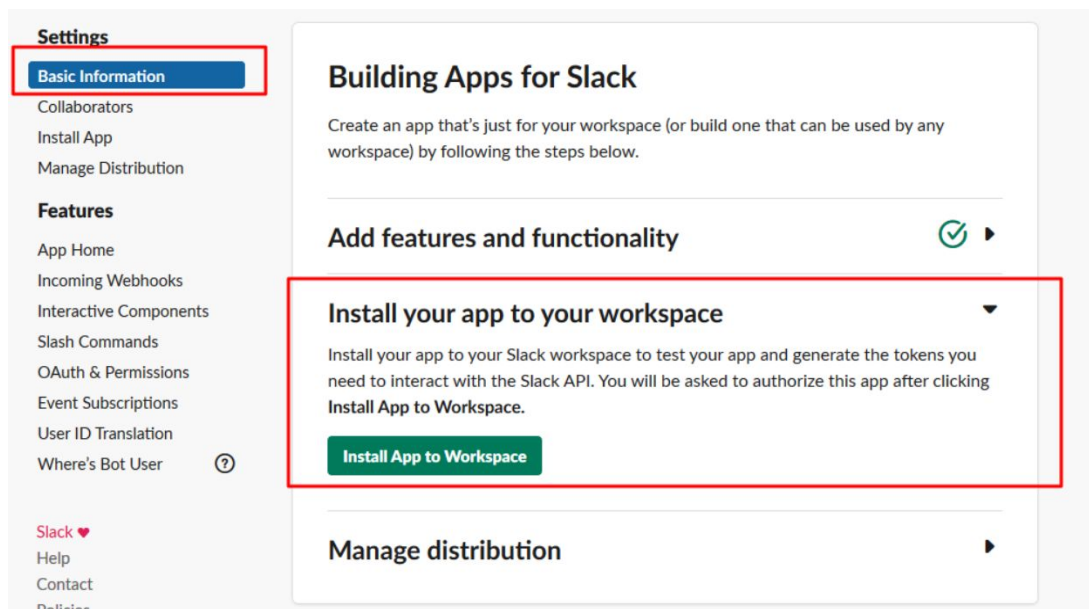
You can add hints such as enabled commands list as a short guide.

In WFH APP, these commands are enabled.

- start -msg [string]
- rest -msg [string]
- rest_end -msg [string]
- task [id] -msg [strings]
- task_end [id] -msg [strings]
- end -msg [string]

And then Click "Save".

3. Add Slack Apps to WorkSpace



4. Register Incoming WebHooks

We need it to post message to channel.

a) Click on link "Incoming Webhooks"

WFH Basic Information

Settings

- Basic Information
- Collaborators
- Install App
- Manage Distribution

Features

- App Home
- Incoming Webhooks
- Interactivity & Shortcuts
- Slash Commands
- OAuth & Permissions
- Event Subscriptions
- User ID Translation
- Where's Bot User

Building Apps for Slack

Create an app that's just for your workspace (or build one that can be used by any workspace) by following the steps below.

Add features and functionality

Choose and configure the tools you'll need to create your app (or review all our documentation).

- Incoming Webhooks**
Post messages from external sources into Slack.
- Interactive Components**
Add components like buttons and select menus to your app's interface, and create an interactive experience for users.
- Slash Commands**
Allow users to perform app actions by typing commands in Slack.
- Event Subscriptions**
Make it easy for your app to respond to activity in Slack.
- Bots**
Allow users to interact with your app through channels and conversations.
- Permissions**
Configure permissions to allow your app to interact with the Slack API.

b) Turn On "Active Incoming Webhooks"

Settings

- Basic Information
- Collaborators
- Install App
- Manage Distribution

Features

- App Home
- Incoming Webhooks**
- Interactivity & Shortcuts
- Slash Commands
- OAuth & Permissions
- Event Subscriptions
- User ID Translation
- Where's Bot User

Activate Incoming Webhooks ☒

Incoming webhooks are a simple way to post messages from external sources into Slack. They make use of normal HTTP requests with a JSON payload, which includes the message and a few other optional details. You can include message attachments to display richly-formatted messages.

Adding incoming webhooks requires a bot user. If your app doesn't have a bot user, we'll add one for you.

Each time your app is installed, a new Webhook URL will be generated.

If you deactivate incoming webhooks, new Webhook URLs will not be generated when your app is installed to your team. If you'd like to remove access to existing Webhook URLs, you will need to [Revoke All OAuth Tokens](#).

Webhook URLs for Your Workspace

To dispatch messages with your webhook URL, send your message in JSON as the body of an `application/json` POST request.

Add this webhook to your workspace below to activate this curl example.

Sample curl request to post to a channel:

```
curl -X POST -H 'Content-type: application/json' --data '{"text": "Hello, World!"}' https://hooks.slack.com/services/TL22BAFPT/QU74P41/kbJT2G1gAQL304VPd5
```

Webhook URL **Channel** **Added By**

https://hooks.slack.com/services/TL22BAFPT/QU74P41/kbJT2G1gAQL304VPd5 Copy	#general	Van Minh Tri Mar 30, 2020
		Van Minh Tri

c) Click "Add new Webhook to Workspace", to add "Slack Apps" to channel

Webhook URLs for Your Workspace




To dispatch messages with your webhook URL, send your [message](#) in JSON as the body of an `application/json` POST request.

Add this webhook to your workspace below to activate this curl example.

Sample curl request to post to a channel:

```
curl -X POST -H 'Content-type: application/json' --data '{"text":"Hello, World!"}' https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5
```

Copy

Webhook URL	Channel	Added By
https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5 Copy	#general	Van Minh Tri Mar 30, 2020 
https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5 Copy	#random	Van Minh Tri Mar 30, 2020 
https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5 Copy	#remote_helper_testing	Van Minh Tri Mar 30, 2020 
Add New Webhook to Workspace		

d) Choose channel that you want post message

**wah is requesting permission to access the
GREVO Co., Ltd. Slack workspace**



Where should wah post?

wah requires a channel to post to as an app

general ▼

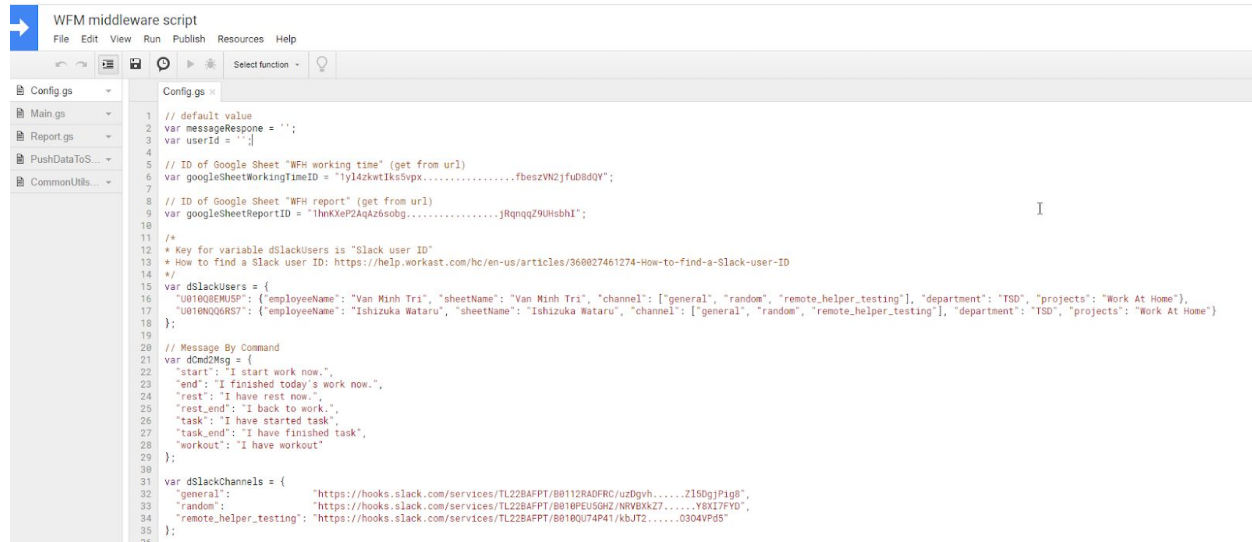
Cancel

Allow

We have done with setting up Slack apps. Now move on to editing WFH APP function.

Set up Config.gs

Come back with Google Apps Script.



We need to config for our environment.

1. Update variable googleSheetWorkingTimeID and googleSheetReportID

We can get that ID in URL of Google Sheet.

2. Update information for variable dSlackUsers

```
var dSlackUsers = {
  "U010Q8EMU5P": {"userName": "Van Minh Tri", "sheetName": "Van Minh Tri",
"channel": ["general", "random", "remote_helper_testing"]},
  ...
};
```

- a) U010Q8EMU5P: This is Slack User Id, Please see how to get it in <https://help.workast.com/hc/en-us/articles/360027461274-How-to-find-a-Slack-user-ID>
- b) userName: Name of User.
- c) sheetName: Name of sheet in google sheet that will create automatic for user.
- d) channel: List channel that apps will post for this user when they type command.

3. Update information for variable dSlackChannels

Come back slack apps setting: <https://api.slack.com/apps>

Copy channel for key and Webhook URL for value.

Webhook URLs for Your Workspace

To dispatch messages with your webhook URL, send your [message](#) in JSON as the body of an `application/json` POST request.

Add this webhook to your workspace below to activate this curl example.

Sample curl request to post to a channel:

```
curl -X POST -H 'Content-type: application/json' --data '{"text":"Hello, World!"}' https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5
```

Copy

Webhook URL	Channel	Added By
https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5 Copy	#general	Van Minh Tri Mar 30, 2020 
https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5 Copy	#random	Van Minh Tri Mar 30, 2020 
https://hooks.slack.com/services/TL22BAFPT/B7U74P41/kbJT2G1gAQL0304VPd5 Copy	#remote_helper_testing	Van Minh Tri Mar 30, 2020 

Add New Webhook to Workspace

Notice

1. If you change anything in Apps Script editor, you need to publish it again. Choose Publish > Deploy as web app...
In popup, choose Project version = "New". And then click button Update.

Deploy as web app

Current web app URL: [Disable web app](#)

Test web app for your [latest code](#).

Project version:

Execute the app as:

You need to authorize the script before distributing the URL.

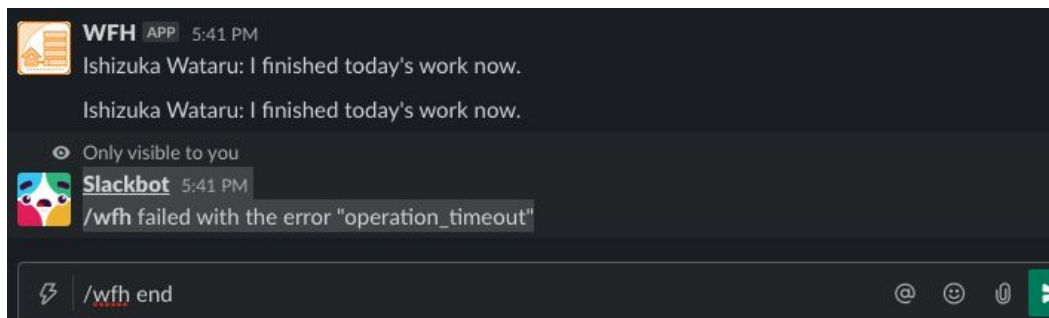
Who has access to the app:

[Help](#)

2. Issues

When you run /wfh commands, Slackbot sometimes returns an error saying "operation_timeout".

This happens because Slack presumes it failed when the command takes over 3 seconds to respond, and then Slackbot sends the message automatically. So, even if that error shows up, the script of WFH APP works correctly.



You'll find your /wfh command runs without a failure in your channel.