

# Slack ❤ Python:

## How we develop SDKs and build bots in Python



# While we're waiting to begin:

1. Open this **slide deck**: [bit.ly/slack-pycon-deck](https://bit.ly/slack-pycon-deck)
2. Join the **Pycon Slack**: [bit.ly/slack-pycon-workshop](https://bit.ly/slack-pycon-workshop)
  - a. **#pycon** and **#bots**
3. Clone the **Pycon demo app**: [github.com/slackapi/pycon-2019](https://github.com/slackapi/pycon-2019)
4. Load the **Slack App Portal**: [api.slack.com/apps](https://api.slack.com/apps)
5. Download **ngrok** and **Python 3.6**



Slack ❤️ Python:  
How we develop SDKs and  
build bots in Python



# Slack's Developer Relations Team



Bear Douglas



Rodney Urquhart



Shane DeWael



Ankur Oberoi



Tomomi Imura



Jim Ray



Jason Roche



Colm Doyle



Taylor Singletary



Matthew Johnston



Krane



Emily Pinkerton



**Rodney Urquhart**  
**@rodneyu215**

Developer Relations  
**Dev Tools - Python**



**Jason Roche**  
**@roach**

Developer Relations  
**Open Source & Community**



**Tomomi Imura**  
**@girlie\_mac**

Developer Relations  
**Developer Education**



**Joe Smith**  
**@yasumoto**

Enterprise Identity  
**Site Reliability Engineer**



**Matt Taylor**  
**(doesn't Tweet)**

Event Marketing  
**Platform Events**



**Elizabeth Kinsey**  
**@eakinsey**

Platform Marketing  
**Developer Community**

# What we do

---



## Docs & Content

Docs, Open API specs, UX guidelines, tutorials, sample code, video content



## Developer Tools & SDKs

Node.js & Python SDKs, Hubot adapter, Steno; SDT; docs, support & outreach



## Enabling internal Integrators

Workshops, App Blueprints, newsletter content, best practice, webinars



## Build community online & in person

Hosting Meetups and Events, and interacting on GitHub and other online spaces

# Why are we here?



We, too, are python developers.

# Our developers are **Python** developers!

---

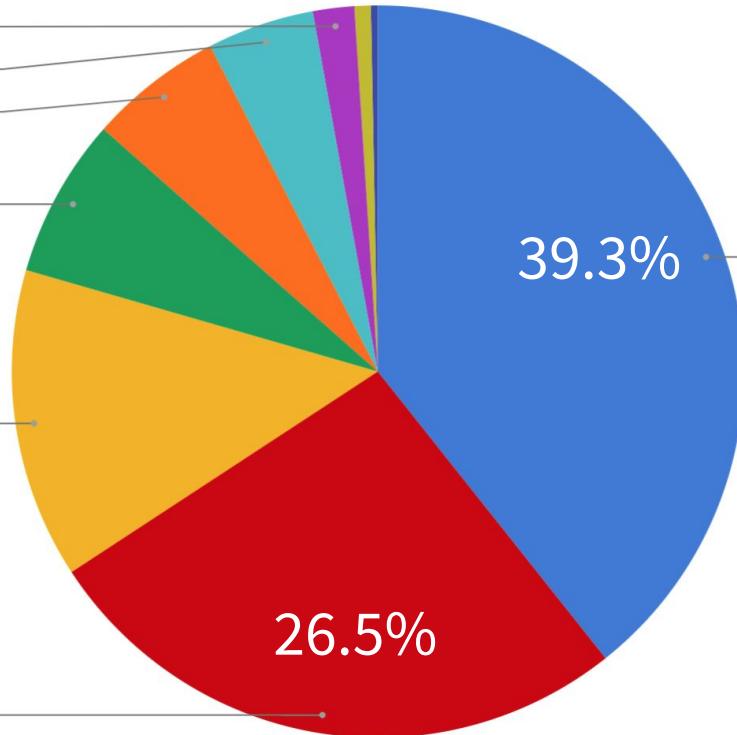
API Usage by SDKs

php  
1.8%  
browser  
4.7%  
ruby  
5.9%

go  
7.0%

java  
13.7%

Node JS



# In today's session, we will:

- Talk about how and why Slack builds Python tools
- Introduce Slack Apps, the Platform APIs and Features
- Teach you how to build a simple Slack App 
- And turn it into a useful tool
- Please feel free to ask questions any time during the workshop!



# External Apps

- Apps which may be installed on any workspace
- Most large SaaS companies have built directory apps

# Internal Apps

- Built to connect internal systems and workflows to Slack
- Customized to connect specific tools inside your org



# The App Directory: Slack apps for every area of work

SALES	DEV TOOLS	COMMUNICATIONS	ANALYTICS	SUPPORT	PRODUCTIVITY
        	     	        	         	        	           
MEDIA & MARKETING	FILE SHARING	DESIGN	HR	FINANCE	IT & SECURITY
       	     	        	        	        	   

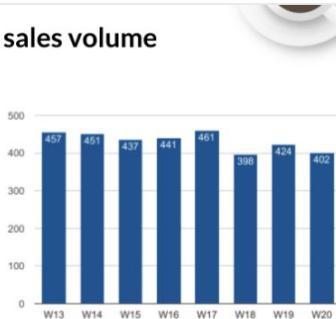
# Apps in the real world: Google Drive

View file previews in Slack

 **Emilie O'Genski** 10:00 AM  
shared this file ▾

 **Coffee mug sales volume + metrics | May 23, 2017**  
Presentation from Google Drive

**Coffee mug sales volume**



Week	Sales Volume
W13	457
W14	451
W15	437
W16	441
W17	461
W18	398
W19	424
W20	402



Get updates on comments, access requests, and new files

 **Google Drive** APP 5:58 PM  
Jane Lorensen [janel@](#) commented on **Summer Retreat Financial Planning**

"Is this going to be enough money for breakfast?  
"Yes, I think this budget is perfect. Thanks!"

 **Google Drive** APP 6:00 PM  
Sarah Smith [@sarah](#) shared **Finance Projections for Summer Retreat**

"You should take a look at this document!"

Easily grant access to files that others want to view

 **Google Drive** APP 6:00 PM  
Sarah Smith [@sarah](#) requested to view **Summer Retreat Financial Planning**

Grant access

# Apps in the real world: Jira

## Ticket previews

Joy Andersen 9:00 AM  
Has anyone looked at WEB-1098?

Jira Cloud APP 9:00AM

**WEB-1098 Adjust borders on rocket ship graphic**

The colour and width of the graphic border needs to be modified to align with the latest guidelines introduced for our launch.

Status	Priority
Awaiting Release	Medium

Type	Assignee
Task	Sarah Mitchell

teaminspace.atlassian.net | May 22nd, 2018

## Attach a message to a Jira ticket

Only visible to

Jira Cloud APP 1 minute ago

Attach message to issue

Marisol Garcia

"Hey @jandersen, small change - can you update the margin between the border and the content to 20px?

Select the site containing the issue you wish to attach this to

Teams in Space

Select the issue that you wish to attach this to

FP-13 As a developer, I can update details...

Are you sure you want to attach this message to FP-13?

Confirm Cancel

## Subscribe a channel to Jira project updates

Jira Cloud APP 9:30AM

Hi Team! Thanks for inviting me to your channel, @Jimmy West!  
I can notify you when things happen in one or more of your Jiras – like this...

**DEV-5: Application server is down**

Our website is down. Displaying 404 error. But don't freak out because this is just a sample issue for you to see

Status
In Progress

Jira

Would you like me to connect this channel to one of your Jiras?

Yes No

## Jira updates in Slack

Jira Cloud APP 11:15AM

Caroline Bennett assigned a Task in Backlog to Jimmy West

**WEB-42 Upgrade app server**

It's time to upgrade our aging server before we have an outage.  
Please reach out to IT to kick-off this process, and then define next steps in this issue so that we're all kept in the loop.

teaminspace.atlassian.net | April 25th, 2018

# Apps in the real world: CONCUR.

Slack as your app's **mobile interface**



Concur Expense Beta APP 2:15 PM

Here's a report to approve (1 of 1)

Report Name

Product Bookclub Lunches

Employee

Liza Gurtin

Report Date

2018-03-21

Amount Due Employee

\$142.48

Claimed Amount

\$142.48

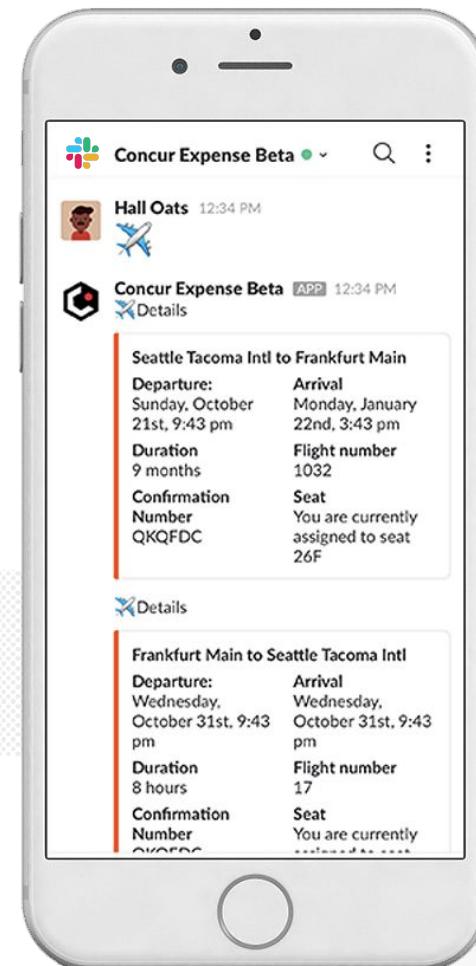
Exceptions

None

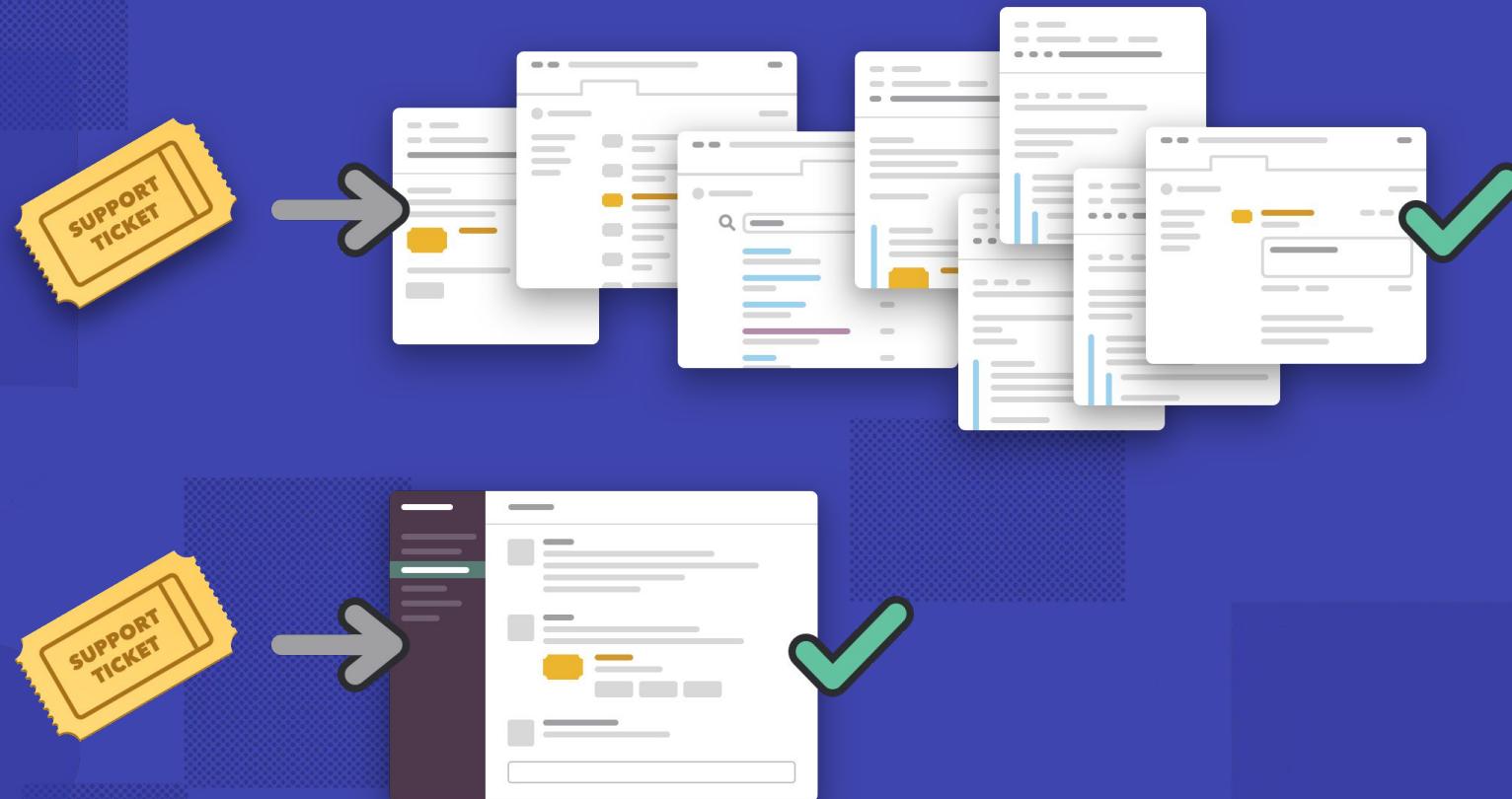
[More Info](#)

[Approve](#)

[Send Back](#)



# Internal Apps: Slack apps for specialized workflows





# /HELPDESK

## Custom App

Simplify IT helpdesk requests

Commands matching "/helpdesk"    tab or ↑ ↓ to navigate    ←→ to select    esc to dismiss

Helpdesk



**/helpdesk** A brief description of your request  
Request support from IT



/helpdesk



### IT Helpdesk



#### Summary

|

#### Request

Choose an option...



#### Location

Choose an option...



#### Is this urgent? (optional)

Choose an option...



#### Description (optional)

3000

[② Learn more about Helpdesk](#)

Cancel

Submit



## TRIAGEBOT

### Custom App

Easily report and track issues



**Brad Slavin** 14:05

At 1pm, the `channels.join` method stopped working. The required argument is `name`, but the requests are failing now with a `channel_not_found` error.

A couple of us have been able to reproduce this. (edited)

Automating status reminders



**Triage Bot** APP 16:00

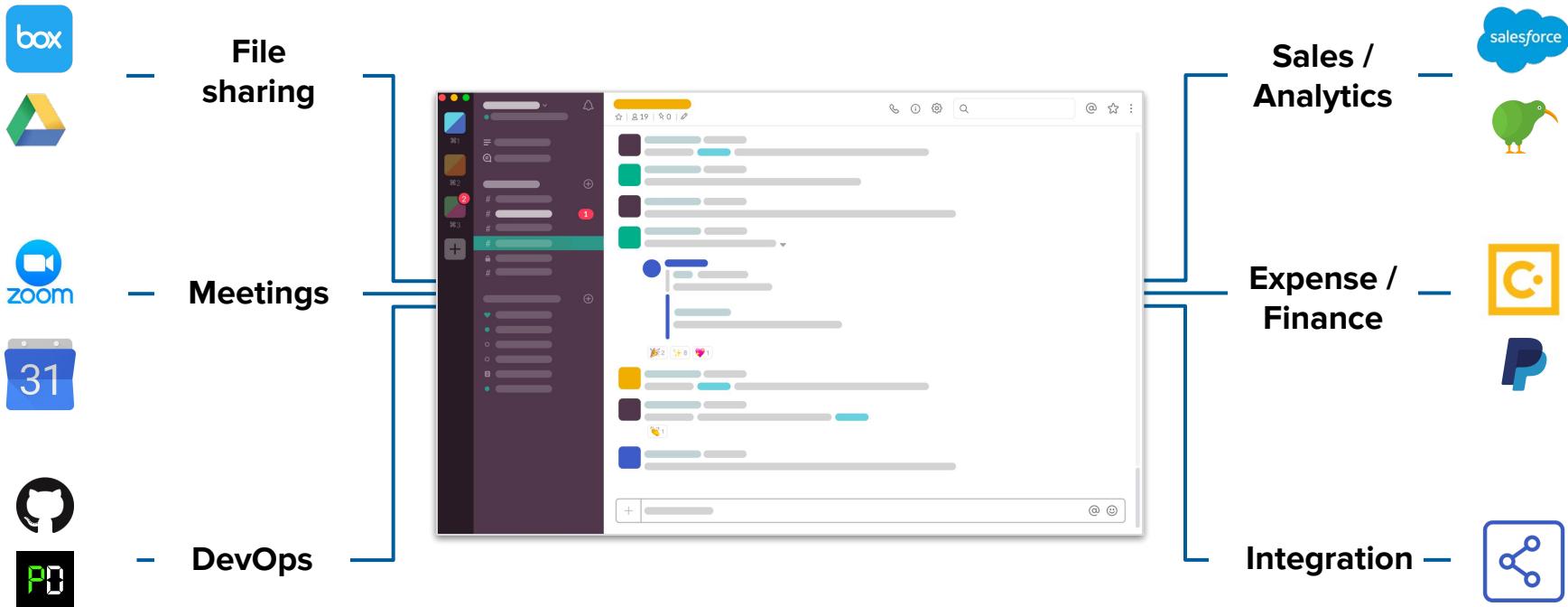
Hi there, @Kefan, @kbomb, and @cristian!

There are 0 triage requests that no one has looked at.

There are 11 triage requests already being looked at

Type `/triage` to see more information.

# Slack: A Collaboration Hub™ - Powered by the Slack Platform



Where your people and applications work together seamlessly.

# Slack Platform Features



# Basic notifications and slash commands

## Notifications

 Slack Team YVR 2:50 PM  
Event starting in 15 minutes:  
**New Hire: Slack 101 [Jamie]**  
Today from 11:30 AM to 12:30 PM at [YVR-Meeting-Room-B](#)  
[Show more ▶](#)

 Honeybadger 5:10 PM  
**ReferenceError** - UI is not defined  
Project Awesome Project Environment production

## Slash commands

 Jimmy West 9:00 AM  
`/baremetrics daily mrr`

 Baremetrics APP 9:00 AM  
**Monthly Recurring Revenue**  
Yesterday: \$90,087  
2 days ago: \$89,587  
Change: 0.56%  


# Message buttons

Book a conference room

 **Jason** 12:22 PM  
meekan what room is free right now?

 **Meekan** APP 12:22 PM  
I have 2 rooms available today at 1:00 PM - 2:00 PM

Death Star	Naboo
Free at 1:00 PM - 4:00 PM	Free at 1:00 PM - 2:00 PM

[Book Death Star](#) [Book Naboo](#) [More options](#)

Approve expenses

 **Sara Tegan** 3:06 PM  
Hello

 **Concur Expense Beta** APP 3:06 PM  
What can I help you with?

[Expenses](#) [Trips](#) [Settings](#)

You can also type `feedback` at anytime to send me ideas, questions and praise.  
E.g. `feedback Thanks for your help, Expense bot!`

**Tip:** If you'd like to add a receipt, upload an image of your receipt as a message  
Hello  @Sara Tegan

Approve file access

 **Google Drive** APP 6:00 PM  
Sarah Smith @sarah requested to view  
**Summer Retreat Financial Planning**

[Grant access](#)

Request more details

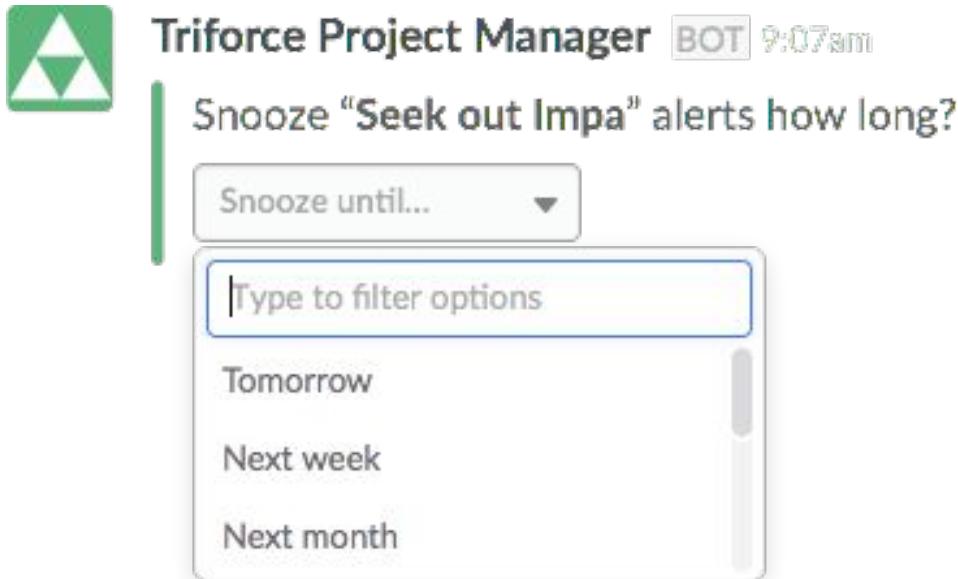
 **ADP Virtual Assistant** APP 9:29 AM

Personal Day Acrl Code -6.58 days	Personal Hrs Acrl Code -8.63 hours
Sick Hrs Acrl Code 179.3 hours	Vacation Day Acrl Code 230.2 days
Vacation Hrs Acrl Code 1497.6 hours	

[Create time off request](#) [View existing requests](#)

# Message menus

---



# Actions and Dialogs

# help-it



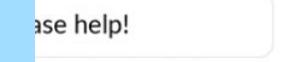
Endo Jac

I lost my



Izzy Mull

I'll make



Submit

# App unfurls



Emilie O'Genski 10:00 AM

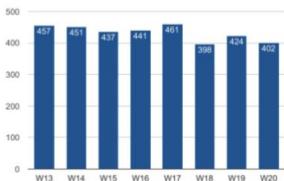
-shared this file ▾



Coffee mug sales volume + metrics | May 23, 2017

Presentation from Google Drive

Coffee mug sales volume



Joy Andersen 9:00 AM

Has anyone looked at WEB-1098?



Jira Cloud APP 9:00AM

**WEB-1098 Adjust borders on rocket ship graphic**

The colour and width of the graphic border needs to be modified to align with the latest guidelines introduced for our launch.

Status

Awaiting Release

Priority

Medium

Type

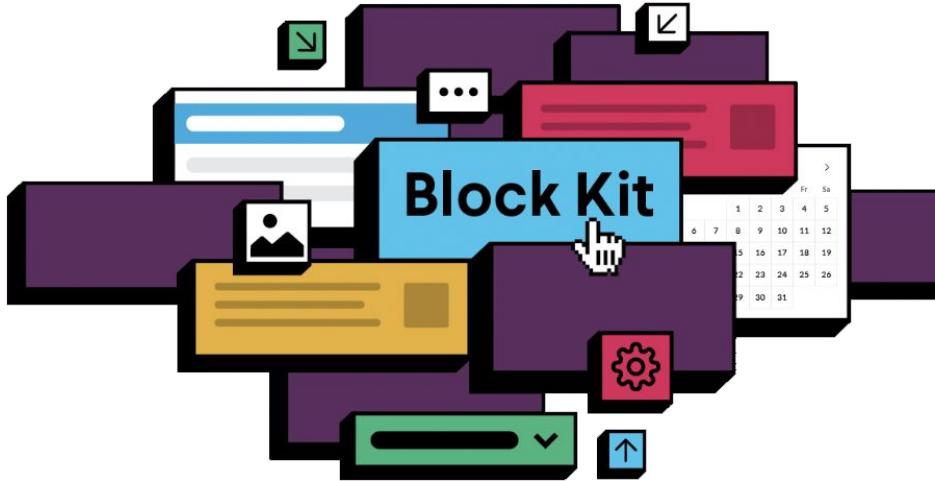
Task

Assignee

Sarah Mitchell



teamsinspace.atlassian.net | May 22nd, 2018



- A **new** message UI framework
- Stackable message UI components
- Block Kit Builder, a visual prototyping tool

## Text

Slackbot APP 10:22 AM  
Welcome to the team!  
We're so thrilled you're here. Before we start, please review and agree to the code of conduct.  
[Read document](#)

## Text collection

Grammar Guide APP 10:22 AM  
Before  
High-quality learning environments are a necessary precondition for facilitation and enhancement of the ongoing learning process.

After  
Children need good schools if they are to learn properly.

## Image

Ripple APP 10:22 AM



"Free mochas this morning at @CoffeeCoLA! Happy Monday, ya'll." Posted by @SaraGifford

## Thumbnail

Nature Rules APP 10:22 AM  
Racoon (Animal)  
The raccoon, sometimes spelled racoon, also known as the common raccoon, North American raccoon, or northern raccoon, is a medium-sized mammal native to North America



## Context collection

Evan Guzman 2:06 PM  
Made a quick fix here, can you review today? - <https://checkitout.blimpagency.com/webapp/pr-9472>

PR #9472  
Refactor profilebuilder log  
3 commits by evangman

Test passed

## Divider

Procurebot APP 10:22 AM  
Hey @paige! Your team has 2 new procurement requests.

Five printers  
Requestor: [@Jimmym](#)  
Comments: We need five printers for buildout of new headquarters.  
Amount: \$6,200.90  
Due: Tomorrow  
[Approve](#) [View](#) [Assign](#)

Content syndication Q3  
Requestor: [@janegoodall](#)  
Comments: Content syndication for integration Q3 campaign.  
Amount: \$3,600  
Due: Wednesday  
[Approve](#) [View](#) [Assign](#)

## Interactive collection

Polls APP 10:22 AM  
Howdy! If you attended yesterday's workshop. We'd really appreciate you answering a few quick questions about it.  
- [@gary](#)

The workshop was a valuable use of my time

[Strongly agree](#) [Agree](#) [Neutral](#) [Disagree](#) [Strongly disagree](#)

## Inline select

My Calendar APP 10:22 AM  
A new #blimpagency event was created by [@sara](#)

Storyboard Review  
10 - 10:30 am  
Acme HQ, Room 5  
 and 3 others

MAY 22

Are you attending?

[RSVP ▾](#)

## Inline overflow

...

[Go to a link](#)

[Take some action](#)

[Take another action](#)

## Date picker

< February 2018 >

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28			

# Block Kit Builder

← → C https://api.slack.com/tools/block-kit-builder?blocks=%5B%7B%22type%22%3A%22section%22%2C%22text%22%3A%22mrkdwn%22%3A%22Hello, Assistant to the Regional Manager Dwight! Michael Scott wants to know where you'd like to take the Paper Company investors to dinner tonight.%22%7D%5D More Info Copy Clear

slack Block Kit Builder Desktop Mobile

## Blocks

- Tt Section
- Image
- i Context
- Actions
- Divider
- Section with Image

Your App APP 1:37 PM

Hello, Assistant to the Regional Manager Dwight! **Michael Scott** wants to know where you'd like to take the Paper Company investors to dinner tonight.

Please select a restaurant:

Farmhouse Thai Cuisine  ★★★★ 1528 reviews  
They do have some vegan options, like the roti and curry, plus they have a ton of salad stuff and noodles can be ordered without meat!! They have something for everyone here

Kin Khao  ★★★★ 1638 reviews  
The sticky rice also goes wonderfully with the caramelized pork belly, which is absolutely melt-in-your-mouth and so soft.

Ler Ros  ★★★★ 2082 reviews  
I would really recommend the Yum Koh Moo Yang - Spicy lime dressing and roasted quick marinated pork shoulder, basil leaves, chili & rice powder.

```
1 [
2   {
3     "type": "section",
4     "text": {
5       "type": "mrkdwn",
6       "text": "Hello, Assistant to the Regional Manager Dwight! *Michael Scott* wants to know where you'd like to take the Paper Company investors to dinner tonight.\n\n*Please select a restaurant:*

Workspace Slack Com... Channel ID C12345 Send to Slack ⓘ


```

# Slack Platform APIs



# Web API

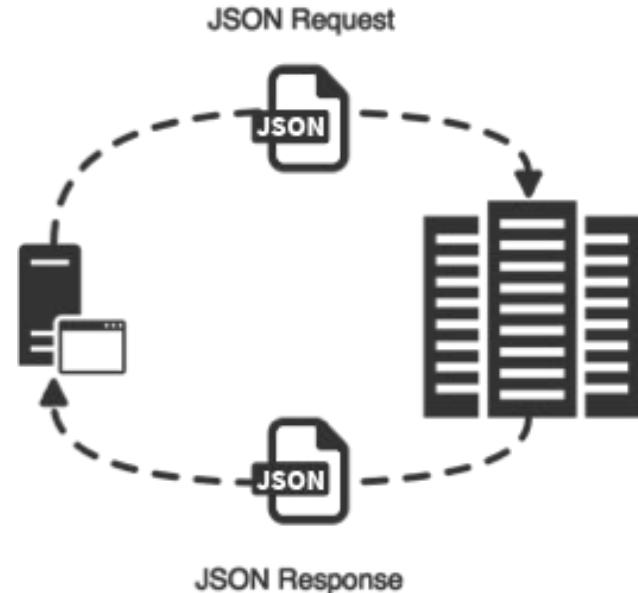
---

Query information, post messages, *change things!* with the Web API.

Use it on the fly for ad-hoc queries or as part of a more complex tapestry of platform features.

It's all HTTP-RPC methods and not quite REST.

**Example:** `chat.postMessage` sends a message



# RTM API

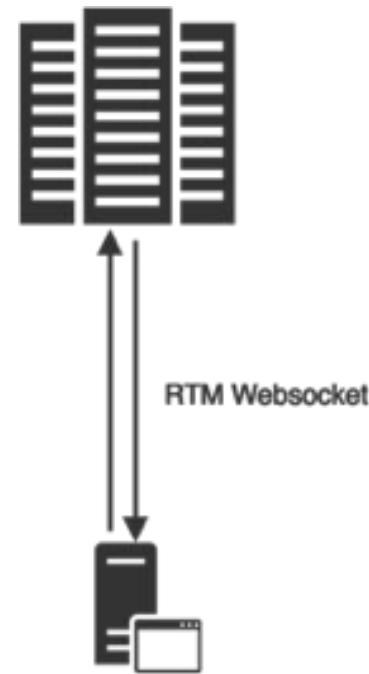
---

The Real Time Messaging API is a WebSocket-based API that allows you to receive events from Slack in real time and send messages as users. It's sometimes referred to as simply the "RTM API".

Doesn't require an HTTP server

Does require an always-on websocket connection

**May not contain many newer Slack Event types, like bot\_mention**



# Interactive & Events API

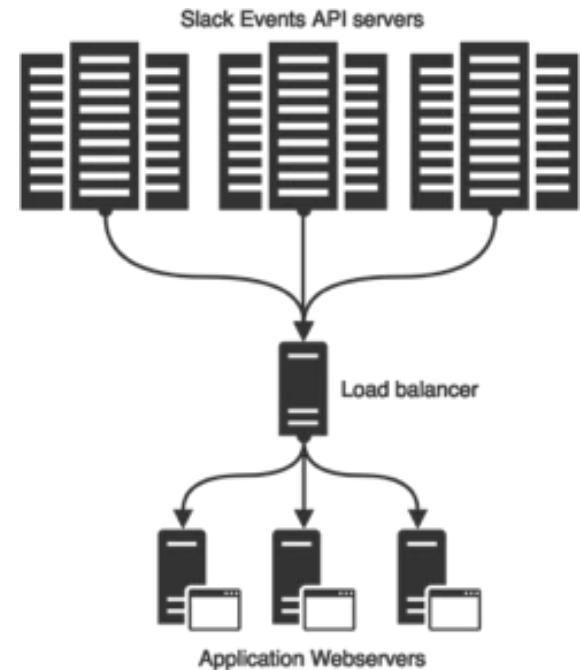
These APIs are subscription models for apps and bots to process & react to activities in Slack.

All you need is a Slack app and a secure place for Slack to send your events.

## Features:

- Events are sent as HTTP requests
- Only subscribe to the events your app needs
- Limit your app's scopes to only those needed
- Most up to date collection of Slack workspace and user events

**Example:** `team_join` is emitted when a new member joins your team



# Slack Platform APIs

	Web API	Interactive & Events API	RTM API
Purpose	The recommended way to send rich, interactive messages in Slack	The recommended way to receive events from Slack	Send basic messages to Slack and receive a stream of events
Protocol	HTTP	HTTP	WebSockets
Send simple messages to Slack	✓		✓
Send interactive messages to Slack	✓		
Receive events from Slack		✓	✓
Primary use cases	Sending messages, taking actions such as creating/modifying channels, uploading files and message attachments, backfilling state	Receiving subscription-based events when users post messages, create/modify channels and files, add reactions to messages, and maintaining state	Sending basic messages/receiving events when there are firewall restrictions and building a Slack client

# Python Slack Tools

The simplicity of Python

The power of Slack



# Python Slack Tools

---

The goal of our Python tools is to make it simple and pleasant for Python developers to create Slack apps.

## **slackclient:**

- Feb 29, 2016 - 1.0 Initial Release
- 16 Releases (Sep 22, 2016 - Feb 28th, 2019)
  - 3 Major Features Added
  - Lots of bug fixes

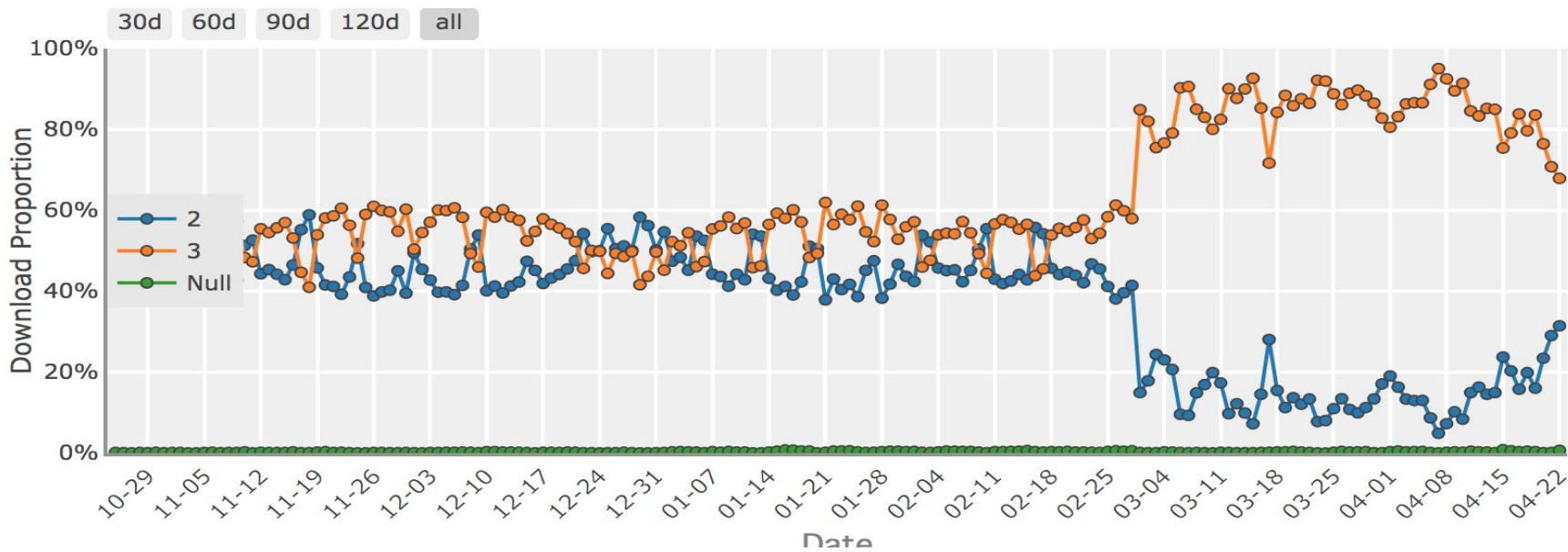
## **slackeventsapi:**

- Dec 18, 2017 - 1.0 Initial Release
- 4 Releases (Jan 19, 2018, Dec 13, 2018)
  - 2 Major Features Added
  - 1 bug fix

# slackclient Redesign Motivation

**Previous decisions led to present challenges:**

- Complexity -> Tough Bugs
- Scope creep -> Poor Maintainability
- Python 2.7 -> Progression 😞



# Seeking feedback from our community

## RFC: Python Slack Client v2.0 #384



RodneyU215 opened this issue on Feb 25 · 24 comments



RodneyU215 commented on Feb 25 • edited

Member



...

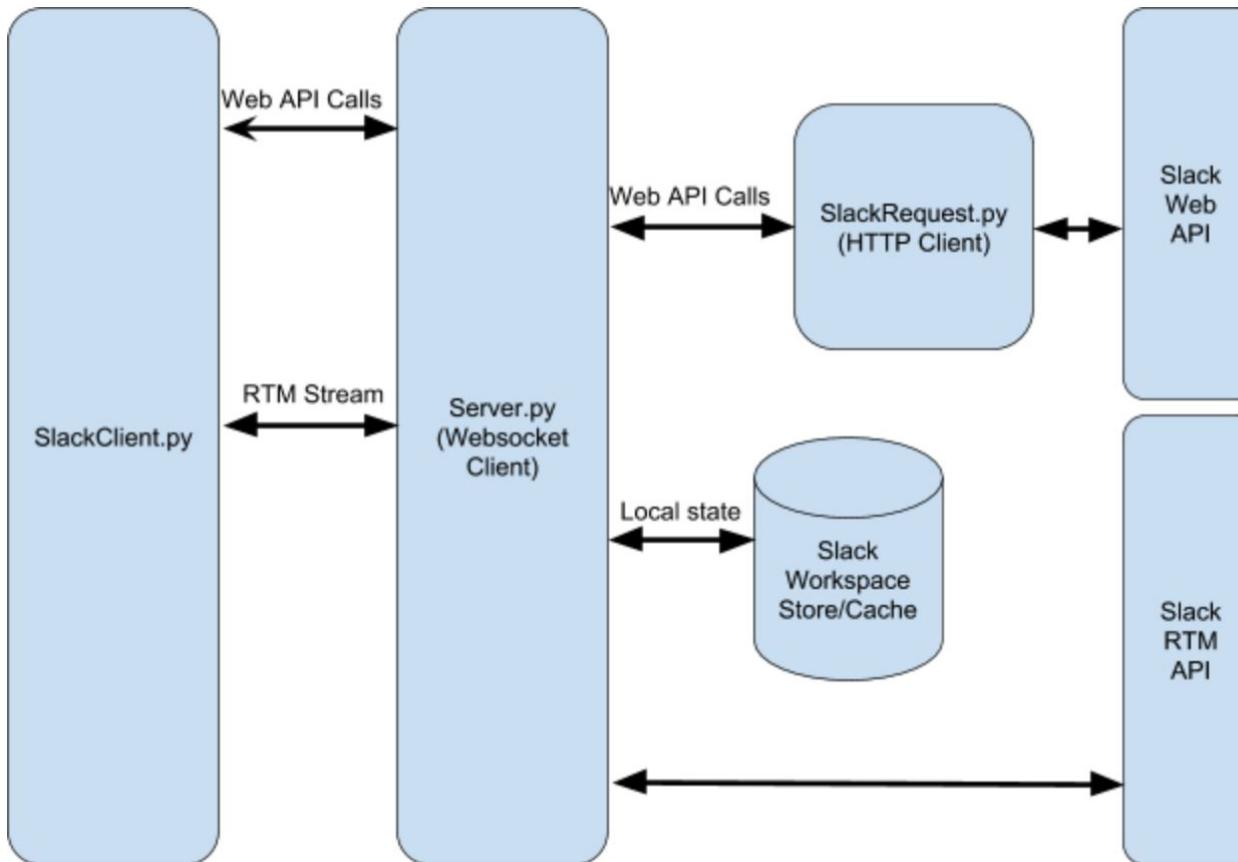
### Abstract

Python developers are an important part of the Slack platform ecosystem. From internal apps for a specific organization or team to publicly distributed apps listed in our App Directory, tools and products built upon our SDKs improve people's working lives.

This issue proposes a redesign of the Slack Python SDK/Client to address several design flaws that surfaced as the scope and complexity of the platform grew. Currently every app built on RTM is required to implement a loop over the stream of events coming in. Web HTTP requests could be simpler to write and more performant when scaled.

This issue proposes that we split up the client into an RTM client and a Web client. We'd like to remove any state not explicitly used for the client's function. As well as make it simpler for developers to interact with the Web API and respond to RTM events.

# Slack Client v2 Architecture



# Seeking feedback from our community



wittekm commented on Mar 5

+ 😊 ...

Any chance we could ask for pep484 annotations on methods like `chat_postMessage`, or is it going to remain generally kwargs-based?



1



ajmacd commented on Mar 11

+ 😊 ...

I had the same question about asynchronous requests while reading the new design. We might consider powering this with a Requests add-on, like [requests-futures](#). Apparently Futures are natively available in Python 3.2 but have been [backported to Python 2](#).

# How we built v2

---

## Infrastructure changes

### Built for Python 3:

- Async support (Optional)
- Type hints and Return types

### Better practices:

- Docstrings for every method.
- Clear Public vs Private interface.
- Deprecation of nonessential client state.

### Streamlined dependencies:

HTTP and Websocket client are now built with “AIOHTTP”.

- Better Proxy and SSL Support
- Built-in async support

## 3 core classes

**WebClient:** The WebClient provides built-in methods for Slack's Web API. These methods act as helpers enabling you to focus more on what you need from Slack and less on how the request is constructed.

**SlackResponse:** These dictionary-like objects of response data also enable you to more easily paginate over list of things in Slack.

**RTMClient:** Completely redesigned with an event-driven architecture, this client allows you to link your application's callbacks to corresponding Slack events.

# Performing actions in Slack and connecting to RTM

---

## Python SlackClient

An easy-to-use client for pulling and posting content into Slack, as well as a robust event-driven websocket connector for use with the RTM API.

```
import slack
```

## Post a message into a Slack channel

```
# Post a message into a Slack channel
client = slack.WebClient('APP_BOT_TOKEN')
client.chat_postMessage(
    channel='#random',
    text='Hello world!'
)
```

## Print incoming message from the RTM API

```
# Connect to the RTM client and print incoming messages
rtm_client = slack.RTMClient('APP_BOT_TOKEN')
@slack.RTMClient.run_on(event='message')
def print_message(**payload):
    print(payload['data']['text'])

rtm_client.start()
```



[github.com/slackapi/python-slackclient](https://github.com/slackapi/python-slackclient)

# Receiving traffic from the Events API

---

## Slack Events API Adapter

A Flask-based library which allows your app to easily receive and process Events API traffic from Slack.

```
from slackeventsapi import SlackEventAdapter
```

## Granular event subscriptions

Event Name	Description
app_mention	Subscribe to only the message events that mention your app or bot
reaction_added	A member has added an emoji reaction to an item

## Print incoming message from the Events API

```
from slackeventsapi import SlackEventAdapter
slack_events_adapter = SlackEventAdapter(
    'SIGNING_SECRET',
    '/slack/events'
)

# Receive message event and print incoming messages
@slack_events_adapter.on('message')
def handle_message(event_data):
    print(event_data['event']['text'])
```



[github.com/slackapi/python-slack-events-api](https://github.com/slackapi/python-slack-events-api)

# Let's build an app!

Greet and React Bot



Test Team ▾

girlie\_mac



Jump to...

All Threads

Channels



# cats

# general

# test-123

# tomomi

# tomomi-01

# tomomi-02

# tomomi-03

Direct Messages



slackbot

girlie\_mac (you)

Testing

## #tomomi-03

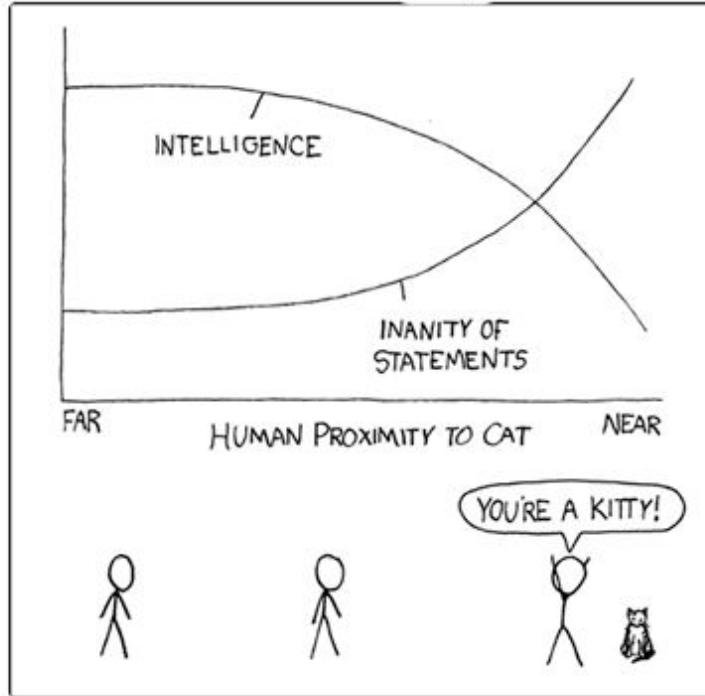
1 | 1 | 0 | Add a topic



Search

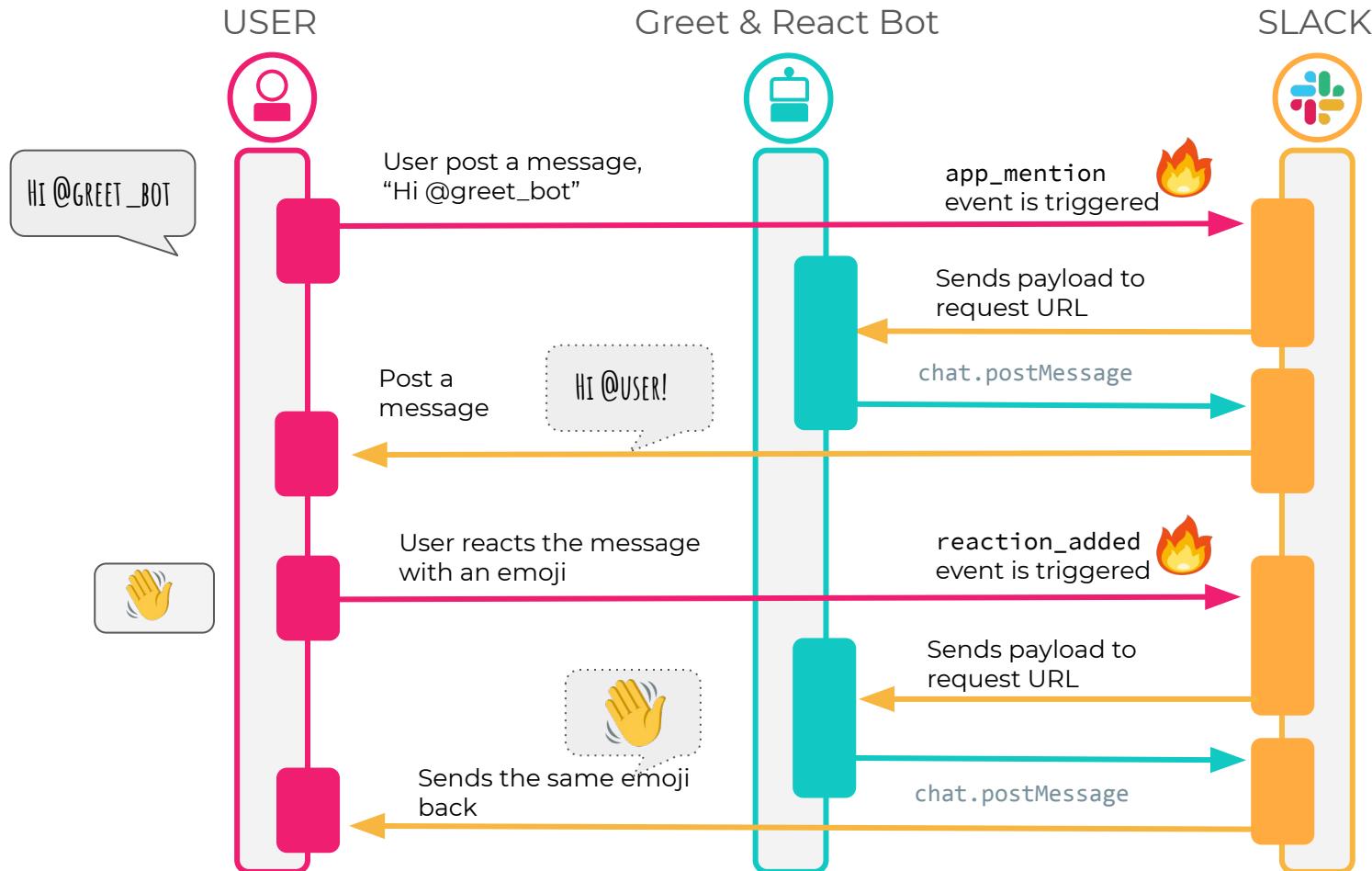


Today



Message #tomomi-03





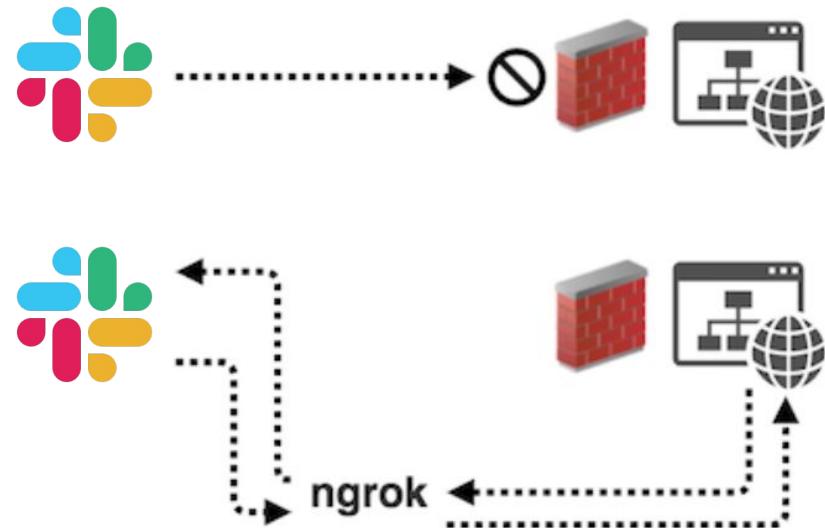
# Configuring and running your app!



# Ngrok - HTTP tunneling

Ngrok is an awesome tool that allows you to expose a local webserver to the internet without having to expose ports to the world. It's especially helpful for developing webhook-style API apps.

[ngrok.com](https://ngrok.com)



**ngrok**

# Ngrok - HTTP tunneling

**Note:** Run this in a separate terminal window from the python app

```
$ ngrok http 3000
```

```
ngrok by @inconsreveable
(Ctrl+C to quit)

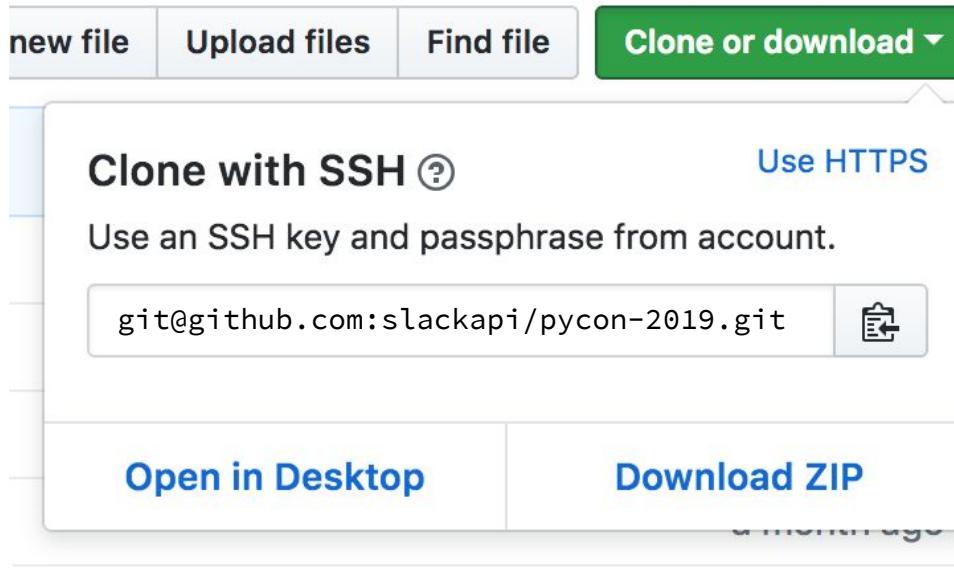
Session Status                online
Account                       Slack (Plan: Basic)

Region                         United States (us)
Web Interface                  http://127.0.0.1:4040
Forwarding                     http://6c6c621f.ngrok.io -> localhost:3000
Forwarding                     https://6c6c621f.ngrok.io -> localhost:3000

Connections                    ttl     opn     rt1     rt5      p50      p90
                                0       0     0.00    0.00    0.00    0.00
```

**ngrok**

# Git clone or download the example app repo:



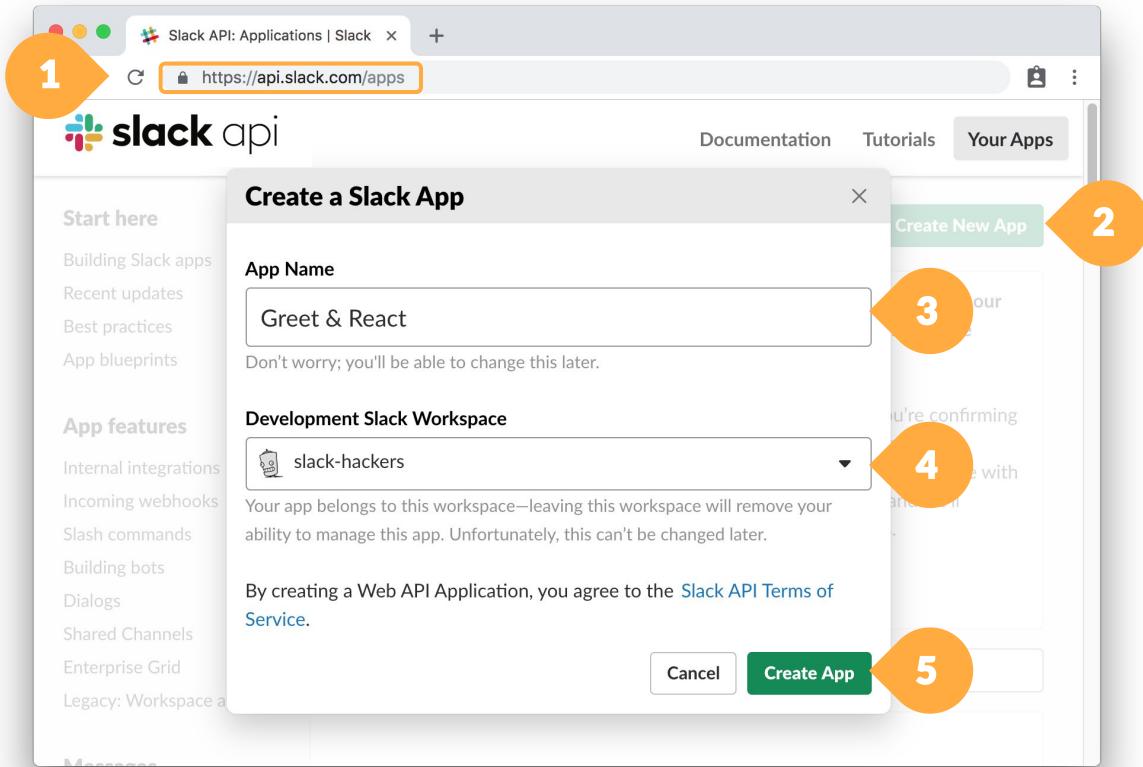
[github.com/slackapi/pycon-2019](https://github.com/slackapi/pycon-2019)



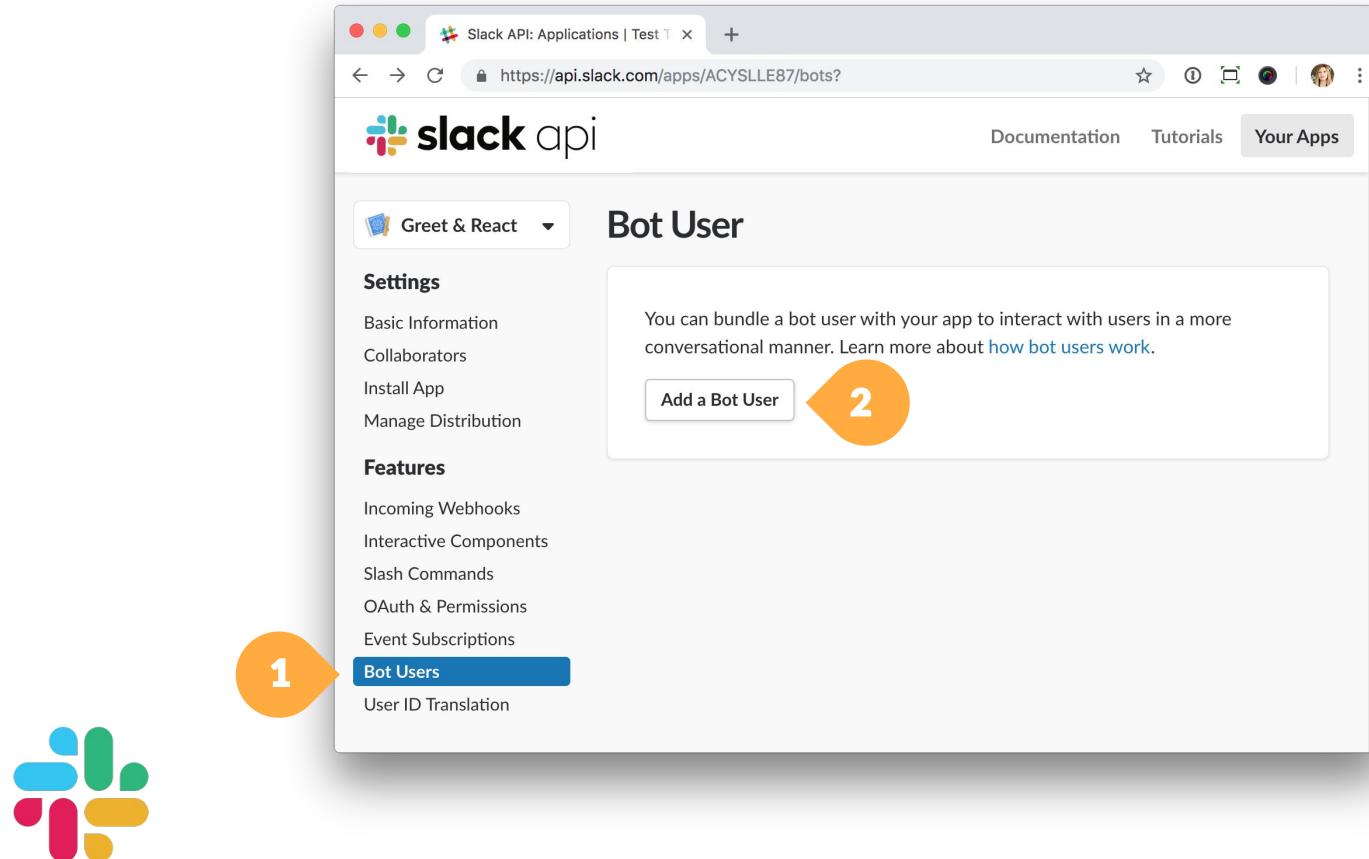
# Create a new Slack App

Go to:  
<https://api.slack.com/apps>

Then click  
“Create New App”



# Add a Bot User (1/2)



2. Click it to add a bot user. (It's a bot but acts as a user!)

# Add a Bot User (2/2)

The screenshot shows a web browser window titled "Slack API: Applications | Test T x". The URL is <https://api.slack.com/apps/ACYSLL87/bots?saved=1>. The page is titled "Bot User". On the left, there's a sidebar with "Settings" and "Features" sections, and a "Bot Users" section which is highlighted with a blue background. The main content area has a heading "You can bundle a bot user with your app to interact with users in a more conversational manner. Learn more about [how bot users work](#)." Below this, there are two input fields: "Display name" containing "greet\_react" and "Default username" also containing "greet\_react". A note below the default username field states: "If this username isn't available on any workspace that tries to install it, we will slightly change it to make it work. Usernames must be all lowercase. They cannot be longer than 21 characters and can only contain letters, numbers, periods, hyphens, and underscores."

3

3. Then give the bot a name at the next screen and save.

(For the group workshop, let's give it a unique name!)



# Install & Get an Access Token (1/3)

The screenshot shows a web browser window titled "Slack API: Applications | Test T" with the URL <https://api.slack.com/apps/AEFQE95C2/install-on-team?>. The page is titled "Install App to Your Team". On the left, there's a sidebar with sections like "Settings" (Basic Information, Collaborators, **Install App**, Manage Distribution) and "Features" (Incoming Webhooks, Interactive Components, Slash Commands, OAuth & Permissions, Event Subscriptions, Bot Users, User ID Translation). A large orange arrow labeled "1" points to the "Install App" button. Another orange arrow labeled "2" points to the green "Install App to Workspace" button. The main content area contains instructions: "Install your app to your Slack workspace to test your app and generate the tokens you need to interact with the Slack API. You will be asked to authorize this app after clicking **Install App to Workspace**".

First, install this app to your workspace, even if your app is not complete.





## Install & Get an Access Token (2/3)

A screenshot of a web browser window titled "Authorize access to your account". The URL in the address bar is <https://test.slack.com/oauth/493233260342.c2612d1c7fa29fb217b64e37db7b...>. The page content includes:

- A header message: "This app was created by a member of your workspace, Test Team."
- An emoji icon of a book with a brain-like pattern next to a green zombie-like head emoji.
- A message: "On Test Team, Greet & React 2 would like to:"
- A section titled "Confirm your identity on Test Team" with a warning message: "⚠ Add a bot user with the username @greet\_react\_2".
- Buttons at the bottom: "Cancel" and "Authorize".

3

Just go ahead and  
install anyway!





# Install & Get an Access Token (3/3)

Now, you should have the tokens generated for you

The screenshot shows the Slack API Applications page for a team named "Test Team". The URL is <https://api.slack.com/apps/ACYSLLE87/install-on-team?>. The page displays "Installed App Settings" for the "Greet & React" app. On the left, there's a sidebar with "Settings" (Basic Information, Collaborators, **Install App**, Manage Distribution) and "Features" (Incoming Webhooks, Interactive Components, Slash Commands, OAuth & Permissions, Event Subscriptions, Bot Users, User ID Translation). The main content area is titled "OAuth Tokens for Your Team", explaining that tokens were automatically generated when the app was installed. It shows two tokens: "OAuth Access Token" (xopb-2147483902-317578579431-439768786787-466e1269e3e2) and "Bot User OAuth Access Token" (xoxb-2147483902-441228832838-O0ltQUQArUIEuKrgAATISkU), each with a "Copy" button. A green "Reinstall App" button is also present.

4



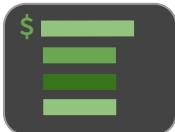
Copy your app's  
Bot Token

# Slack App Credentials

---

Set your **SLACK\_BOT\_TOKEN** env variable

```
$ export SLACK_BOT_TOKEN="<YOUR BOT TOKEN>"
```





# Get Your App Credentials

The screenshot shows a web browser window titled "Slack API: Applications | Test" with the URL <https://api.slack.com/apps/ACYSLL87/general?>. On the left, a sidebar menu includes "Greet & React", "Settings" (selected), and "Basic Information". Under "Settings", there are links for "Collaborators", "Install App", and "Manage Distribution". Under "Basic Information", there are links for "Incoming Webhooks", "Interactive Components", "Slash Commands", "OAuth & Permissions", and "Event Subscriptions". The main content area is titled "App Credentials" and contains the following information:

App ID	Date of App Creation
ACYSLL87	September 20, 2018

**Client ID:** 2147483902.440904694279

**Client Secret:** \*\*\*\*\* Show Regenerate

You'll need to send this secret along with your client ID when making your [oauth.access](#) request.

**Signing Secret:** \*\*\*\*\* Show Regenerate

Slack signs the requests we send you using this secret. Confirm that each request.

Discard Changes Save Changes

A large orange circle with the number "1" is positioned to the left of the sidebar menu. A large orange circle with the number "2" is positioned to the right of the "Signing Secret" section.



2  
Copy your app's  
**Signing Secret**



# Slack App Credentials

---

Set your **SLACK\_SIGNING\_SECRET** env variable

```
$ export SLACK_BOT_TOKEN="<YOUR BOT TOKEN>"  
$ export SLACK_SIGNING_SECRET="<YOUR SECRET>"
```



# Ngrok - HTTP tunneling

Copy your ngrok HTTPS URL

```
ngrok by @inconshreveable  
(Ctrl+C to quit)
```

```
Session Status
```

```
online
```

```
Account
```

```
Slack (Plan: Basic)
```

```
Region
```

```
United States (us)
```

```
Web Interface
```

```
http://127.0.0.1:4040
```

```
Forwarding
```

```
http://6c6c621f.ngrok.io -> localhost:3000
```

```
Forwarding
```

```
https://6c6c621f.ngrok.io -> localhost:3000
```

```
Connections
```

	ttl	opn	rt1	rt5	p50	p90
	0	0	0.00	0.00	0.00	0.00

```
https://6c6c621f.ngrok.io -> localhost:3000
```

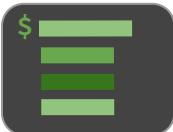
ngrok

# Run your app

---

Set your **SLACK\_SIGNING\_SECRET** env variable

```
$ export SLACK_BOT_TOKEN=<YOUR BOT TOKEN>
$ export SLACK_SIGNING_SECRET=<YOUR SECRET>
$ python app.py
```



# Enable Events

Go to Event Subscriptions



The screenshot shows the Slack API Applications interface. The URL in the browser is <https://api.slack.com/apps/ACYSLLE87/event-subscriptions>. The page title is "Event Subscriptions". On the left, there's a sidebar with "Greet & React" selected under "Settings" and "Event Subscriptions" selected under "Features". The main content area has a heading "Enable Events" with a green "On" toggle switch. Below it, a paragraph explains event subscriptions and a "Request URL" input field contains "https://laksj8s7.ngrok.io/slack/events". A "Change" button is next to the URL. A note below says "We'll send HTTP POST requests to this URL when events occur." At the bottom are "Discard Changes" and "Save Changes" buttons.

1 Enable it!

2 Enter your Request URL

Your server URL (<https://your-repo.glitch.me/slack/events>)

# Event handler URL (app.py)

```
# Create an instance of SlackEventAdapter, passing in our Flask server so it can bind the
# Slack specific routes. The `endpoint` param specifies where to listen for Slack event
traffic.

slack_events_adapter = SlackEventAdapter(
    slack_signing_secret,
    endpoint="/slack/events"
)

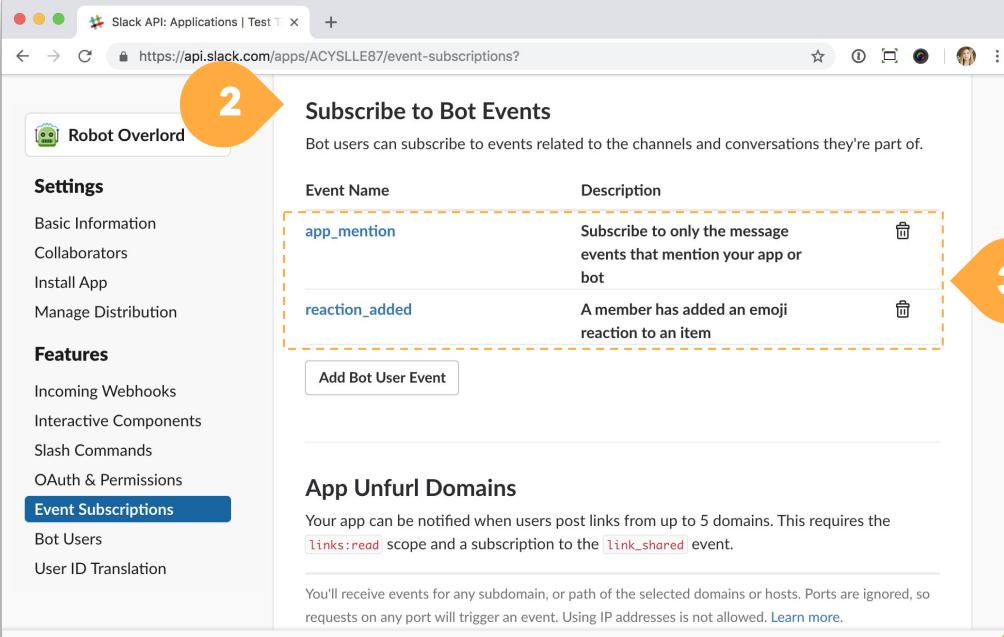

Must match the Request URL you specified!
e.g. If your Request URL is https://example.com/fo, then this should be '/fo'
```

```
# Create an instance of SlackClient for your bot to make Web API requests,
# passing your app's Bot Token.
slack_client = slack.WebClient(token=os.environ["SLACK_BOT_TOKEN"])
```



# Subscribing Events

Scroll down to  
**Subscribe to Bot  
Events**



The screenshot shows the Slack API Applications Test interface. On the left, a sidebar lists various app settings and features. The 'Event Subscriptions' option is highlighted with a blue background and white text. The main content area is titled 'Subscribe to Bot Events' and contains a table of event subscriptions. Two events are listed: 'app\_mention' and 'reaction\_added'. Both events have a trash can icon to their right. At the bottom of the table is a button labeled 'Add Bot User Event'. Below the table, there's a section titled 'App Unfurl Domains' with instructions on how to receive notifications for links posted from specific domains.

Event Name	Description
app_mention	Subscribe to only the message events that mention your app or bot
reaction_added	A member has added an emoji reaction to an item

**Discard Changes** **Save Changes**

1

2

3

4

3. Add these two events, **app\_mention** & **reaction\_added**

4. Click **Save Changes**



# Events: app\_mention

```
@slack_events_adapter.on("app_mention")
def handle_message(event_data):
    # Get the message metadata from the `event` portion of the request.
    message = event_data["event"]

    # We're logging out the payload so you can see it's anatomy
    print(json.dumps(message, indent=4, sort_keys=True))
```

Event is fired!

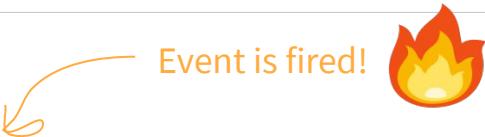


```
# If the incoming message contains "hi", then respond with a "Hello" message
# the `subtype` check filters out messages from other bot users.
if message.get("subtype") is None and "hi" in message['text']:
    channel = message["channel"]
    message = "Hello <@%s>! :tada:" % message["user"]
    slack_client.chat_postMessage(channel=channel, text=message)
```



# Events: reaction\_added

```
# Example reaction emoji echo
@slack_events_adapter.on("reaction_added")
def reaction_added(event_data):
    event = event_data["event"]
    emoji = event["reaction"]
    channel = event["item"]["channel"]
    text = ":%s:" % emoji
    slack_client.chat_postMessage(channel=channel, text=message)
```



Test Team ▾

girlie\_mac



Jump to...

All Threads

Channels



# cats

# general

# test-123

# tomomi

# tomomi-01

# tomomi-02

# tomomi-03

Direct Messages



slackbot

girlie\_mac (you)

Testing

## #tomomi-03

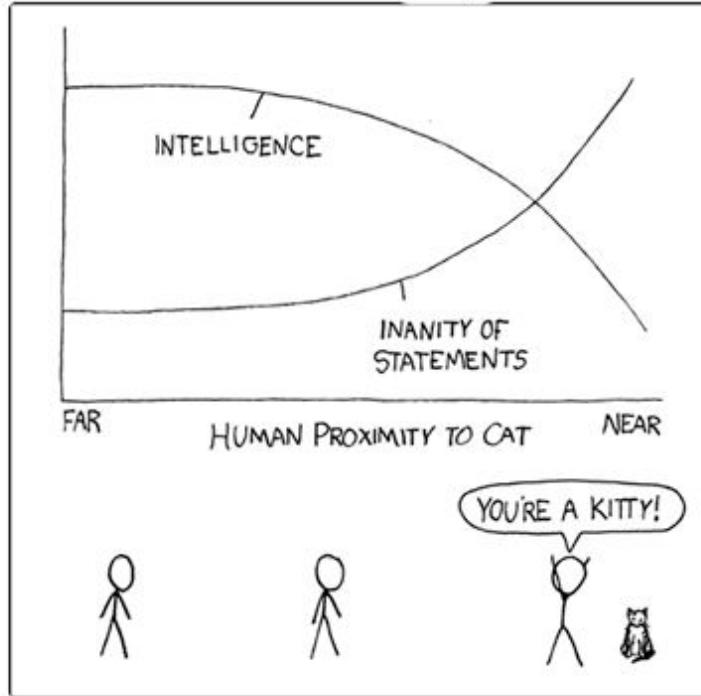
1 | 1 | 0 | Add a topic



Search



Today



Message #tomomi-03



# What's next?

Turn your app into one of these useful tools:

- ★ A Single-channel Reacji-Channeler 😺

Share messages to a particular channel based on reaction events



- ★ A Simple Triage bot 💊

Listen to incoming traffic and highlight important messages



## TRIAGEBOT

### Custom App

Easily report and track issues



**Brad Slavin** 14:05

At 1pm, the `channels.join` method stopped working. The required argument is `name`, but the requests are failing now with a `channel_not_found` error.

A couple of us have been able to reproduce this. (edited)

Automating status reminders



**Triage Bot** APP 16:00

Hi there, @Kefan, @kbomb, and @cristian!

There are 0 triage requests that no one has looked at.

There are 11 triage requests already being looked at

Type `/triage` to see more information.

## More information:

- **Slack API Docs:** [api.slack.com](http://api.slack.com)
- **Python SDKs:** [slack.dev/python-slackclient](http://slack.dev/python-slackclient)
- **Other SDKs and Tools:** [slack.dev](http://slack.dev)
- **App Blueprints and Code Samples:**
  - [glitch.com/@slack-blueprints](http://glitch.com/@slack-blueprints)
  - [github.com/slackapi](http://github.com/slackapi)
- **Slack Community Meetups:**
  - <http://slackcommunity.com>

 @slackapi



# Now, go build an app!

**Take it a step further, then one more after that.  
Stop by our booth to show us what you've  
made to enter to win some special swag!**



We'll announce and contact winners Saturday at  
lunch - good luck!

