
flask_slack Documentation

Release 0.1.5

VeryCB

January 30, 2016

1	Quickstart	3
2	Installation	5
3	API Reference	7
3.1	Slack	7
3.2	Exceptions	8
	Python Module Index	9

Flask-Slack is a Flask extension which makes it easy to interact with Slack slash commands.

Quickstart

You can register your Slack slash command to a method as follows.

```
from flask_slack import Slack

slack = Slack(app)

@slack.command('your_command', token='your_token',
              team_id='your_team_id', methods=['POST'])
def your_method(**kwargs):
    text = kwargs.get('text')
    return slack.response(text)
```

All registered methods are dispatched through `slack.dispatch` method. You can connect it to an endpoint which you wish Slack robots to request with.

```
app.add_url_rule('/', view_func=slack.dispatch)
```

Installation

You can install Flask-Slack with pip:

```
$ pip install flask-slack
```

Or, with setuptools easy_install in case you didn't have pip:

```
$ easy_install flask-slack
```

API Reference

3.1 Slack

class flask_slack.**Slack** (*app=None*)

command (*command, token, team_id=None, methods=['GET'], **kwargs*)

A decorator used to register a command. Example:

```
@slack.command('your_command', token='your_token',
               team_id='your_team_id', methods=['POST'])
def your_method(**kwargs):
    text = kwargs.get('text')
    return slack.response(text)
```

Parameters

- **command** – the command to register
- **token** – your command token provided by slack
- **team_id** – optional. your team_id provided by slack. You can also specify the “TEAM_ID” in app configuration file for one-team project
- **methods** – optional. HTTP methods which are accepted to execute the command
- **kwargs** – optional. the optional arguments which will be passed to your register method

dispatch ()

Dispatch http request to registered commands. Example:

```
slack = Slack(app)
app.add_url_rule('/', view_func=slack.dispatch)
```

init_app (*app=None*)

Initialize application configuration

response (*text, response_type='ephemeral', attachments=None*)

Return a response with json format

Parameters

- **text** – the text returned to the client

- **response_type** – optional. When *in_channel* is assigned, both the response message and the initial message typed by the user will be shared in the channel. When *ephemeral* is assigned, the response message will be visible only to the user that issued the command.
- **attachments** – optional. A list of additional messages for rich response.

validate (*command, token, team_id, method*)

Validate request queries with registered commands

Parameters

- **command** – command parameter from request
- **token** – token parameter from request
- **team_id** – team_id parameter from request
- **method** – the request method

3.2 Exceptions

class flask_slack.**SlackError** (*msg*)

Base Slack Exception

__init__ (*msg*)

Return error message to user

Parameters **msg** – the error message to return

f

Flask-Script, [3](#)
flask_slack, [7](#)

Symbols

`__init__()` (`flask_slack.SlackError` method), 8

C

`command()` (`flask_slack.Slack` method), 7

D

`dispatch()` (`flask_slack.Slack` method), 7

F

Flask-Script (module), 1

`flask_slack` (module), 7, 8

I

`init_app()` (`flask_slack.Slack` method), 7

R

`response()` (`flask_slack.Slack` method), 7

S

`Slack` (class in `flask_slack`), 7

`SlackError` (class in `flask_slack`), 8

V

`validate()` (`flask_slack.Slack` method), 8