

---

# **pullover Documentation**

***Release 1.0.0***

**George Brighton**

**Jul 06, 2020**



---

## Contents

---

<b>1</b>	<b>Send a message</b>	<b>3</b>
<b>2</b>	<b>Querying the response</b>	<b>5</b>
<b>3</b>	<b>API Documentation</b>	<b>7</b>
3.1	Low-level API . . . . .	7
3.2	Exceptions . . . . .	9
<b>4</b>	<b>Indices</b>	<b>11</b>
	<b>Python Module Index</b>	<b>13</b>
	<b>Index</b>	<b>15</b>



The simplest Pushover API wrapper for Python, release v1.0.0.



---

## Send a message

---

To send a message, simply:

```
>>> import pullover
>>> pullover.send('message', 'user', 'app')
```

You can additionally pass any parameters accepted by *Message*'s initialiser.

`pullover.send(body, user_key, app_token, **kwargs)`  
Send a message to a user from an application.

### Parameters

- **body** (*str*) – The body of the message to send.
- **user\_key** (*str*) – The user key to send to.
- **app\_token** (*str*) – The application token to send from.
- **kwargs** – Additional keyword arguments to pass to *Message*'s initialiser.

**Returns** A message send response instance.

**Return type** *SendResponse*





## CHAPTER 2

---

### Querying the response

---

`pullover.send()` returns a `SendResponse` object, from which the status, request ID and any errors can be retrieved.

```
>>> response = pullover.send('message', 'user', 'app')
>>> response.ok
True
>>> response.id
5042853c-402d-4a18-abcb-168734a801de
>>> response.status
1
>>> response.errors
[]
```

If you'd like to use exceptions, `raise_for_status()` will raise either a `ClientSendError` or `ServerSendError` if `ok` is False.

**class** `pullover.message.SendResponse(response)`

Represents the Pushover API's response to a message send request.

**ok**

Find whether the response indicates the message was successfully sent.

**Returns** True if it was, false otherwise.

**Return type** `bool`

**raise\_for\_status()**

Raise an appropriate exception given this response.

**Raises** `SendError` – If this response indicates a request failed.



## 3.1 Low-level API

The high-level API exposes the same functionality as the low-level API in a simpler way, hiding the creation and interaction of various objects.

### 3.1.1 Applications

```
class pullover.Application (token)
    Encapsulates a Pushover application token, and signs requests with it.
    __init__ (token)
        Initialise a new application.
        Parameters token (str) – The application token.
```

### 3.1.2 Users

```
class pullover.User (key)
    Encapsulates a Pushover user key, and signs requests with it.
    __init__ (key)
        Initialise a new user.
        Parameters key (str) – The user key.
```

### 3.1.3 Messages

```
class pullover.Message (body, title=None, timestamp=None, url=None, url_title=None, priority=0)
    Represents a Pushover message.
    HIGH = 1
```

`LOW = -1`

`LOWEST = -2`

`NORMAL = 0`

`__init__(body, title=None, timestamp=None, url=None, url_title=None, priority=0)`

Initialise a new message.

#### Parameters

- **body** (*str*) – The contents of the message.
- **title** (*str*) – The message heading. If not provided, the name of the sending application will be shown.
- **timestamp** (*datetime.datetime*) – The message datetime. Defaults to now.
- **url** (*str*) – A supplementary URL to show underneath the message.
- **url\_title** (*str*) – The title for the URL above. Requires URL be set.
- **priority** (*int*) – The message priority, e.g. *HIGH*. Defaults to *NORMAL*.

Raises **ValueError** – If a URL title is provided, but no URL.

`prepare(application, user)`

Package up this message with a sending application and user, ready for sending.

#### Parameters

- **application** (*Application*) – The application to send the message from.
- **user** (*User*) – The user to send the message to. All devices will receive it.

**Returns** A prepared message object.

**Return type** *PreparedMessage*

`send(application, user, timeout=3, retry_interval=5, max_tries=5)`

Send this message to a user, making it originate from a given application. This method guarantees not to throw any exceptions.

#### Parameters

- **application** (*Application*) – The application to send the message from.
- **user** (*User*) – The user to send the message to. All devices will receive it.
- **timeout** (*float*) – The number of seconds to allow for each request to Pushover. Defaults to 3s.
- **retry\_interval** (*float*) – The amount of time to wait between requests. Defaults to 5s. Note, this is the *minimum recommended by Pushover*.
- **max\_tries** (*int*) – The number of attempts to make before giving up. Defaults to 5. Set this to 1 to disable back-off.

**Returns** The result of the send attempt.

**Return type** *SendResponse*

## Prepared messages

If you want your *Message*, *Application* and *User* creation logic to be separate from your sending logic, prepared messages may help. These are essentially just a tuple containing all three objects, that you can call *send()* on when ready.

**class** pullover.**PreparedMessage** (*message, application, user*)

A message together with its sending application and receiving user.

**send** (*\*\*kwargs*)

Send this prepared message.

**Parameters** *kwargs* – Additional parameters to pass to *Message.send()*.

**Returns** The result of the send attempt.

**Return type** *SendResponse*

## 3.2 Exceptions

**class** pullover.**PulloverError**

The abstract base class of all errors raised by pullover.

**class** pullover.**SendError**

Derived instances of this abstract class are raised by *raise\_for\_status()* if a request was not successful.

**class** pullover.**ClientSendError** (*status, errors*)

Represents a message send error where we're at fault.

**class** pullover.**ServerSendError** (*response*)

Represents a message send error where Pushover is experiencing issues.



## CHAPTER 4

---

### Indices

---

- `genindex`
- `modindex`
- `search`





**p**

pullover, 3



## Symbols

`__init__()` (*pullover.Application method*), 7  
`__init__()` (*pullover.Message method*), 8  
`__init__()` (*pullover.User method*), 7

## A

*Application (class in pullover)*, 7

## C

*ClientSendError (class in pullover)*, 9

## H

*HIGH (pullover.Message attribute)*, 7

## L

*LOW (pullover.Message attribute)*, 7  
*LOWEST (pullover.Message attribute)*, 8

## M

*Message (class in pullover)*, 7

## N

*NORMAL (pullover.Message attribute)*, 8

## O

*ok (pullover.message.SendResponse attribute)*, 5

## P

*prepare()* (*pullover.Message method*), 8  
*PreparedMessage (class in pullover)*, 9  
*pullover (module)*, 3  
*PulloverError (class in pullover)*, 9

## R

*raise\_for\_status()*  
    (*pullover.message.SendResponse method*),  
    5

## S

*send()* (*in module pullover*), 3  
*send()* (*pullover.Message method*), 8  
*send()* (*pullover.PreparedMessage method*), 9  
*SendError (class in pullover)*, 9  
*SendResponse (class in pullover.message)*, 5  
*ServerSendError (class in pullover)*, 9

## U

*User (class in pullover)*, 7