

---

# **aiobearychat Documentation**

***0.1.0***

**mozillazg**

**8 27, 2017**



---

## Contents

---

<b>1</b>	<b>aiobearychat</b>	<b>3</b>
1.1	Features . . . . .	3
1.2	Installation . . . . .	3
1.3	Usage . . . . .	4
1.4	Credits . . . . .	4
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>API (OpenAPI)</b>	<b>7</b>
3.1	. . . . .	7
<b>4</b>	<b>Contributing</b>	<b>9</b>
4.1	Types of Contributions . . . . .	9
4.2	Get Started! . . . . .	10
4.3	Pull Request Guidelines . . . . .	11
4.4	Tips . . . . .	11
<b>5</b>	<b>Credits</b>	<b>13</b>
5.1	Development Lead . . . . .	13
5.2	Contributors . . . . .	13
<b>6</b>	<b>Changelog</b>	<b>15</b>
6.1	0.1.0 (2017-08-20) . . . . .	15
<b>7</b>	<b>API Reference</b>	<b>17</b>
7.1	OpenAPI . . . . .	17
<b>8</b>	<b>Indices and tables</b>	<b>19</b>



Contents:



An async [BearyChat](#) API library for Python

- Free software: MIT license
- Documentation: <https://aiobearychat.readthedocs.org>
- GitHub: <https://github.com/mozillazg/aiobearychat>
- PyPI: <https://pypi.python.org/pypi/aiobearychat>
- Python version: 3.5, 3.6

## Features

- I/O [BearyChat](#) Python SDK
- I/O http aiohttp, tornado, ...)
- OpenAPI

## Installation

At the command line:

```
$ pip install aiobearychat[aiohttp]
```

## Usage

### OpenAPI

```
import aiohttp

from aiobearychat.openapi.aiohttp import OpenAPI

async def main(token):
    async with aiohttp.ClientSession() as session:
        api = OpenAPI(session, token=token)
        response = await api.user.list()
        print(response.json())
```

## Credits

This package was created with [Cookiecutter](#) and the [mozillazg/cookiecutter-pypackage](#) project template.



## CHAPTER 2

---

### Installation

---

At the command line:

```
$ pip install aiobearychat
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv aiobearychat  
$ pip install aiobearychat
```



## CHAPTER 3

---

### API (OpenAPI)

---

```
import aiohttp

from aiobearychat.openapi.aiohttp import OpenAPI

async def main(token):
    async with aiohttp.ClientSession() as session:
        api = OpenAPI(session, token=token)
        response = await api.user.list()
        print(response.json())
```

OpenAPI aiobearychat.openapi.OpenAPI aiohttp http

api aiobearychat.openapi.OpenAPI



Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

### Types of Contributions

#### Report Bugs

Report bugs at <https://github.com/mozillazg/aio-bearychat/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

#### Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

#### Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

## Write Documentation

aiobearychat could always use more documentation, whether as part of the official aiobearychat docs, in docstrings, or even on the web in blog posts, articles, and such.

## Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/mozillazg/aiobearychat/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## Get Started!

Ready to contribute? Here's how to set up *aiobearychat* for local development.

1. Fork the *aiobearychat* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/aiobearychat.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv aiobearychat
$ cd aiobearychat/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 aiobearychat tests
$ python setup.py test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

## Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 3.5 and 3.6. Check [https://travis-ci.org/mozillazg/aiobearychat/pull\\_requests](https://travis-ci.org/mozillazg/aiobearychat/pull_requests) and make sure that the tests pass for all supported Python versions.

## Tips

To run a subset of tests:

```
$ py.test tests/test_aiobearychat.py
```





## CHAPTER 5

---

### Credits

---

#### Development Lead

- [mozillazg](#) <[mozillazg101@gmail.com](mailto:mozillazg101@gmail.com)>

#### Contributors

None yet. Why not be the first?



## CHAPTER 6

---

### Changelog

---

#### **0.1.0 (2017-08-20)**

- First release on PyPI.
- OpenAPI API.



## CHAPTER 7

---

### API Reference

---

#### OpenAPI



## CHAPTER 8

---

### Indices and tables

---

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