SATELLITE

BERLIN 2019

WORKSHOP DAY

Integrate

GitHub platform: what, why, and how?

Introduction - Erika Kato

- Started at GitHub in September, 2018.
- Partner Engineering manager
- Previous experiences in embedded systems and online technologies

Discussion Topics

- What is the GitHub platform?
- Why do we have it?
- How do we use it?

Poll Time!

Workshop lab

- Create an out-of-office responder when issues are assigned to you.

What is the GitHub Platform?

- "GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 36 million developers."

- Exposes the ability for external applications to be built, to enable a different set of use cases.

It could be for a single user…

Or to provide a solution to many

Why do we have it?

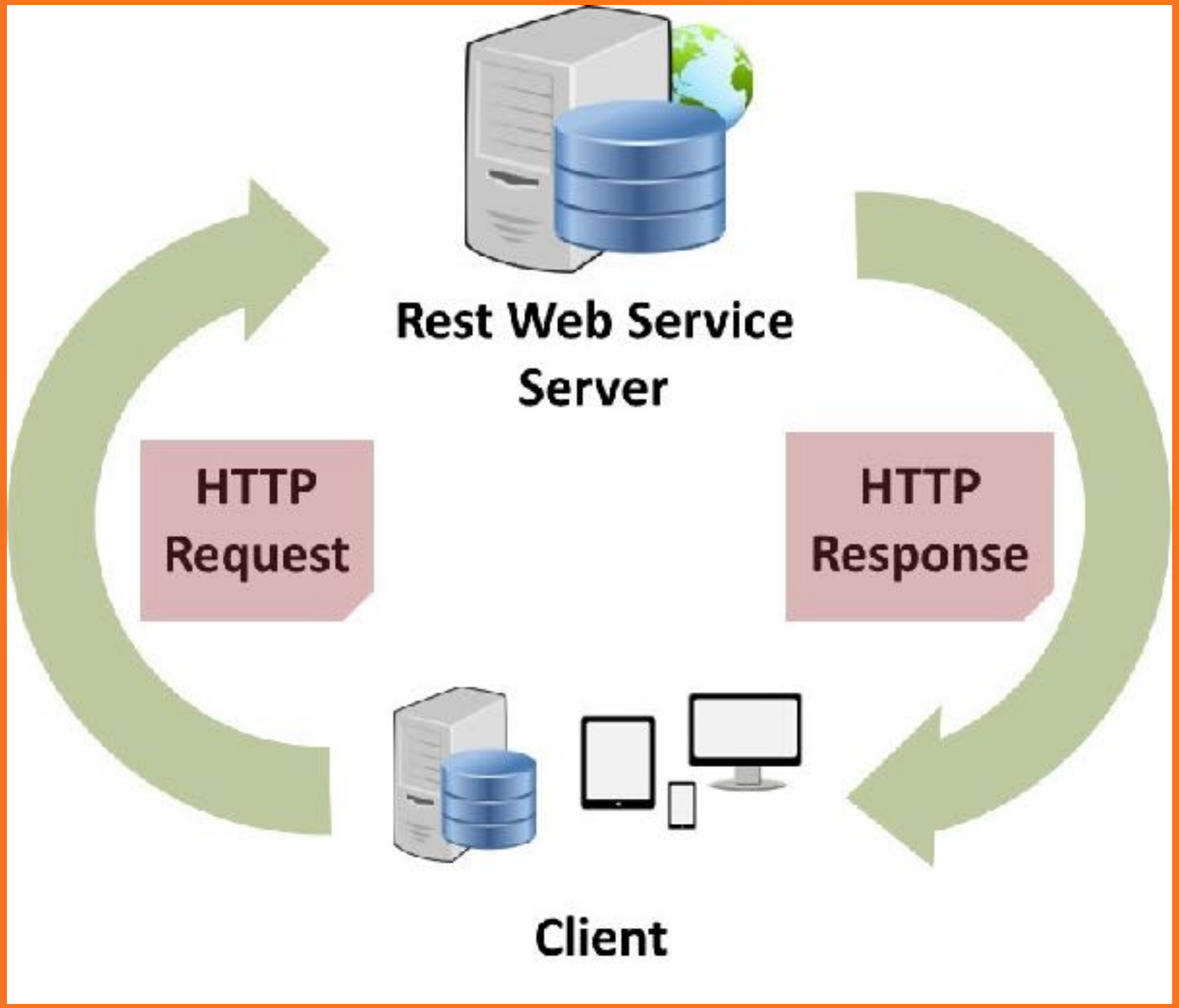
Why do we have it?

- Increased innovation
- Allows other businesses to build and integrate tools with GitHub.
- Allows users and organizations to build custom tools to fit their unique needs.
- Allows users to programmatically collect and analyze GitHub data.
    - GitHub BigQuery public data set
        - Find trends

How do we use it?

What is Available

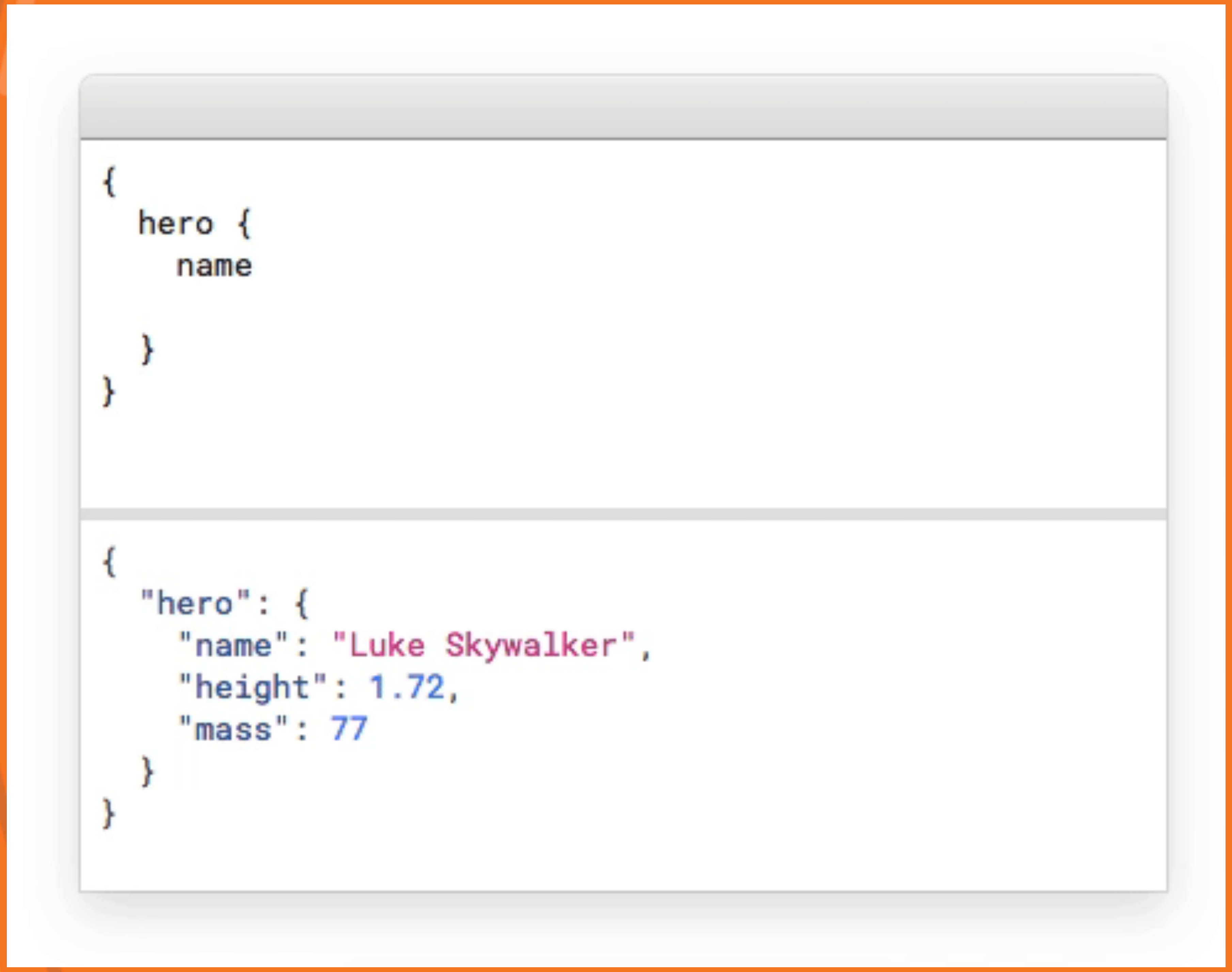
- REST API



- HTTP requests to GET, PUT, POST, and DELETE data.
- There's also HEAD, PATCH, CONNECT, OPTIONS, and TRACE
- Stateless protocol

What is Available

- GraphQL



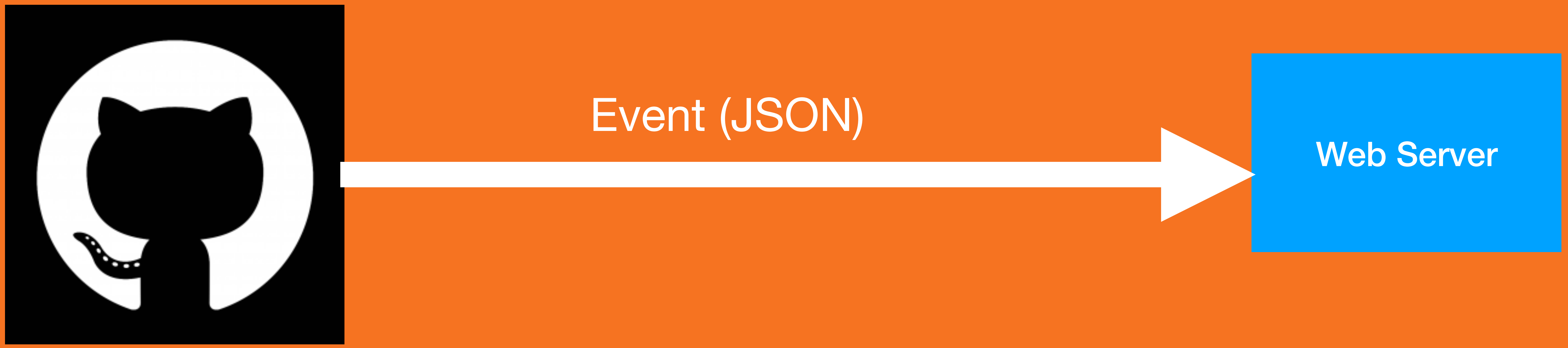
```
{
  hero {
    name

  }
}
```

```
{
  "hero": {
    "name": "Luke Skywalker",
    "height": 1.72,
    "mass": 77
  }
}
```

- Allows the power to ask for exactly what you need and nothing more.

What is Available

- Webhooks
  - A way to provide real-time information
  - “Push” as opposed to “poll”
  - 44 events available



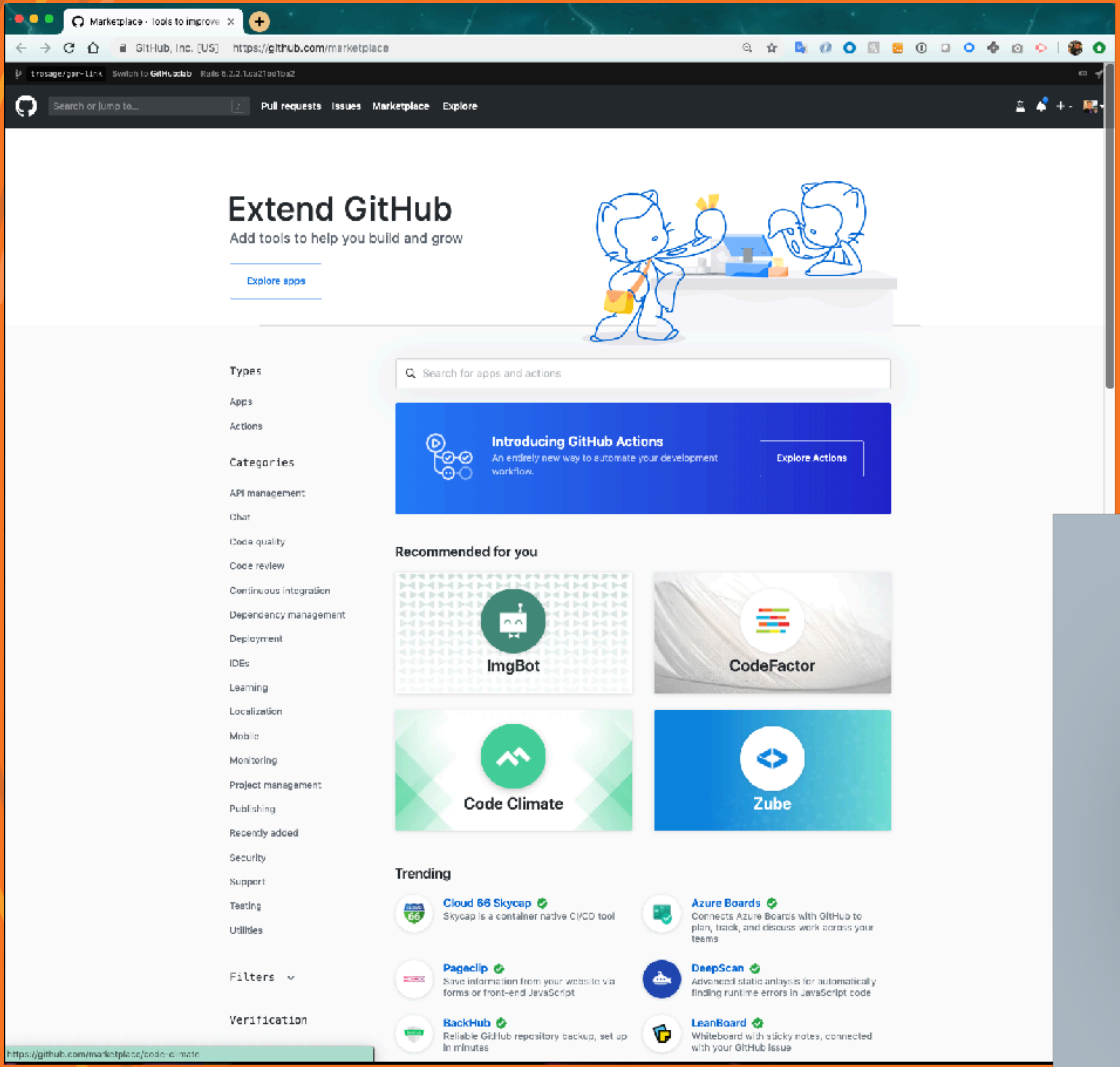
What is Available

- GitHub Actions ✨
  - Automate workflow
  - Powerful workflows to supercharge your repos
  - Executed on demand as autoscaled containers

GitHub Apps

- Fine-grained permissions - simpler installation to organizations and user repos
- Short-lived tokens - better for security
- Centralized webhooks
- Dedicated API rate limits

Marketplace



Marketplace · Tools to improve

GitHub, Inc. [US] https://github.com/marketplace

Pull requests Issues Marketplace Explore

Search or jump to...

Extend GitHub

Add tools to help you build and grow

Explore apps

Types

Apps

Actions

Categories

API management

Chat

Code quality

Code review

Continuous integration

Dependency management

Deployment

IDEs

Learning

Localization

Mobile

Monitoring

Project management

Publishing

Recently added

Security

Support

Testing

Utilities

Filters

Verification

Search for apps and actions

Introducing GitHub Actions

An entirely new way to automate your development workflow.

Explore Actions

Recommended for you

ImgBot

CodeFactor

Code Climate

Zube

Trending

Cloud 66 Skycap Skycap is a container native CI/CD tool

Azure Boards Connects Azure Boards with GitHub to plan, track, and discuss work across your teams

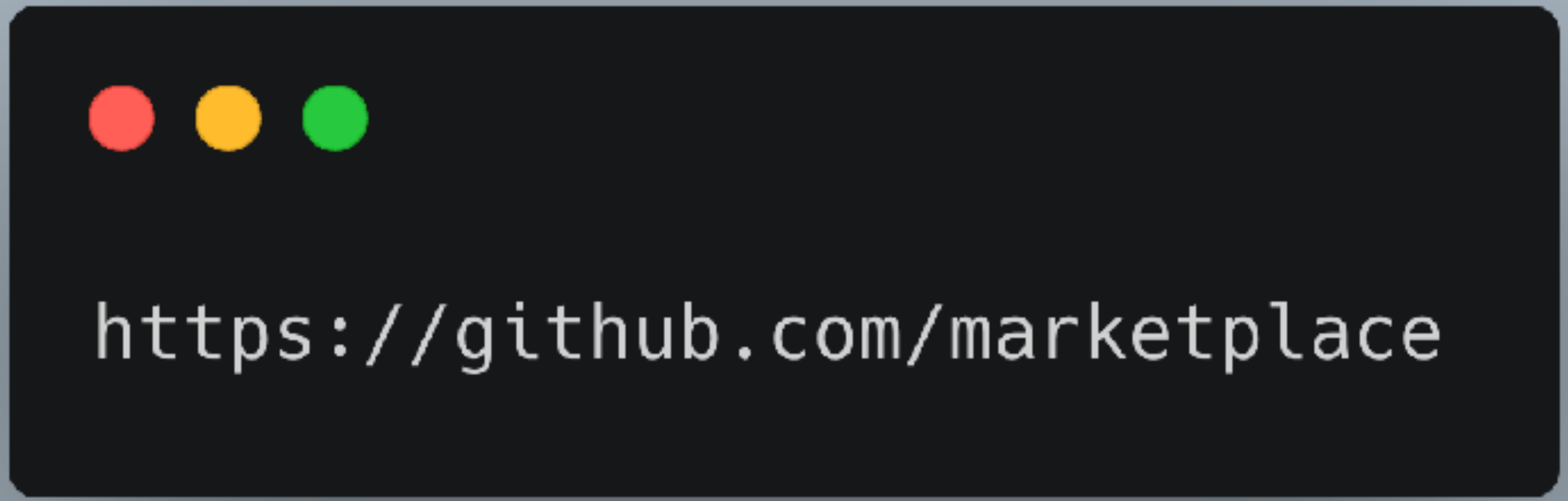
Pageclip Save information from your website via forms or front-end JavaScript

DeepScan Advanced static analysis for automatically finding runtime errors in JavaScript code

BackHub Reliable GitHub repository backup, set up in minutes

LeanBoard Whiteboard with sticky notes, connected with your GitHub issue

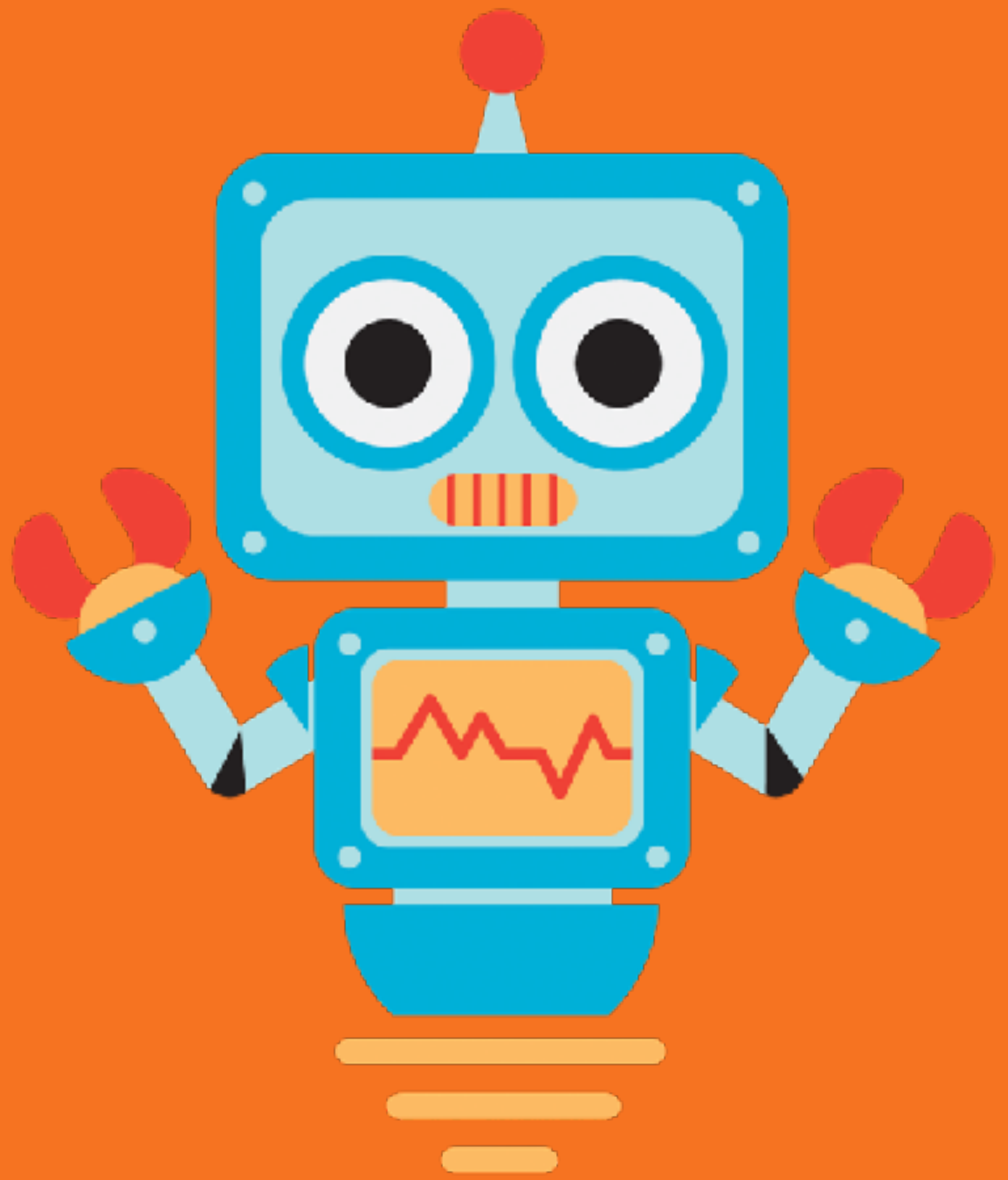
https://github.com/marketplace/code-climate

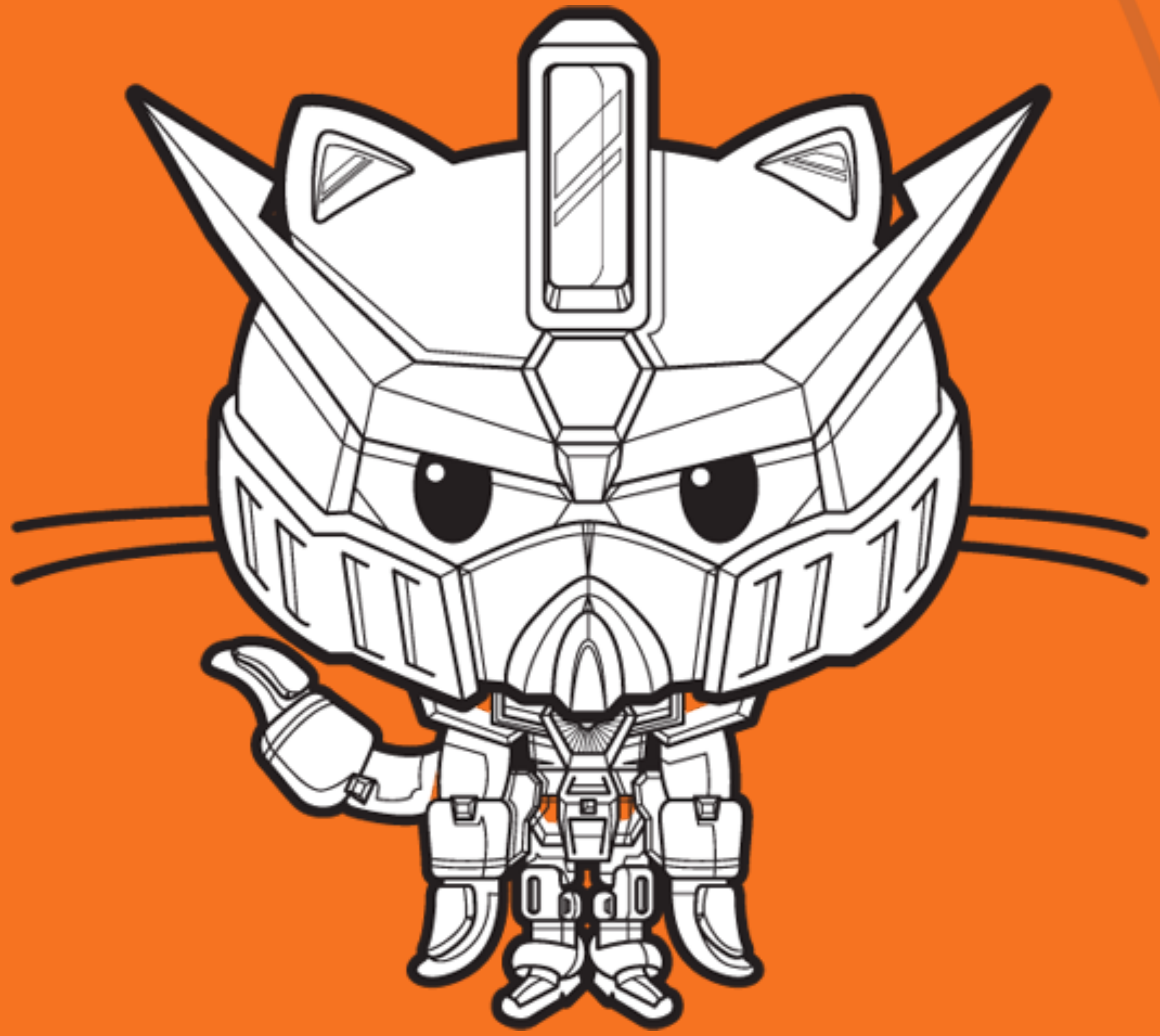


```
https://github.com/marketplace
```

Octokit and Probot

- Octokit has many flavors
- Probot removes a lot of common setups needed when building a GitHub app

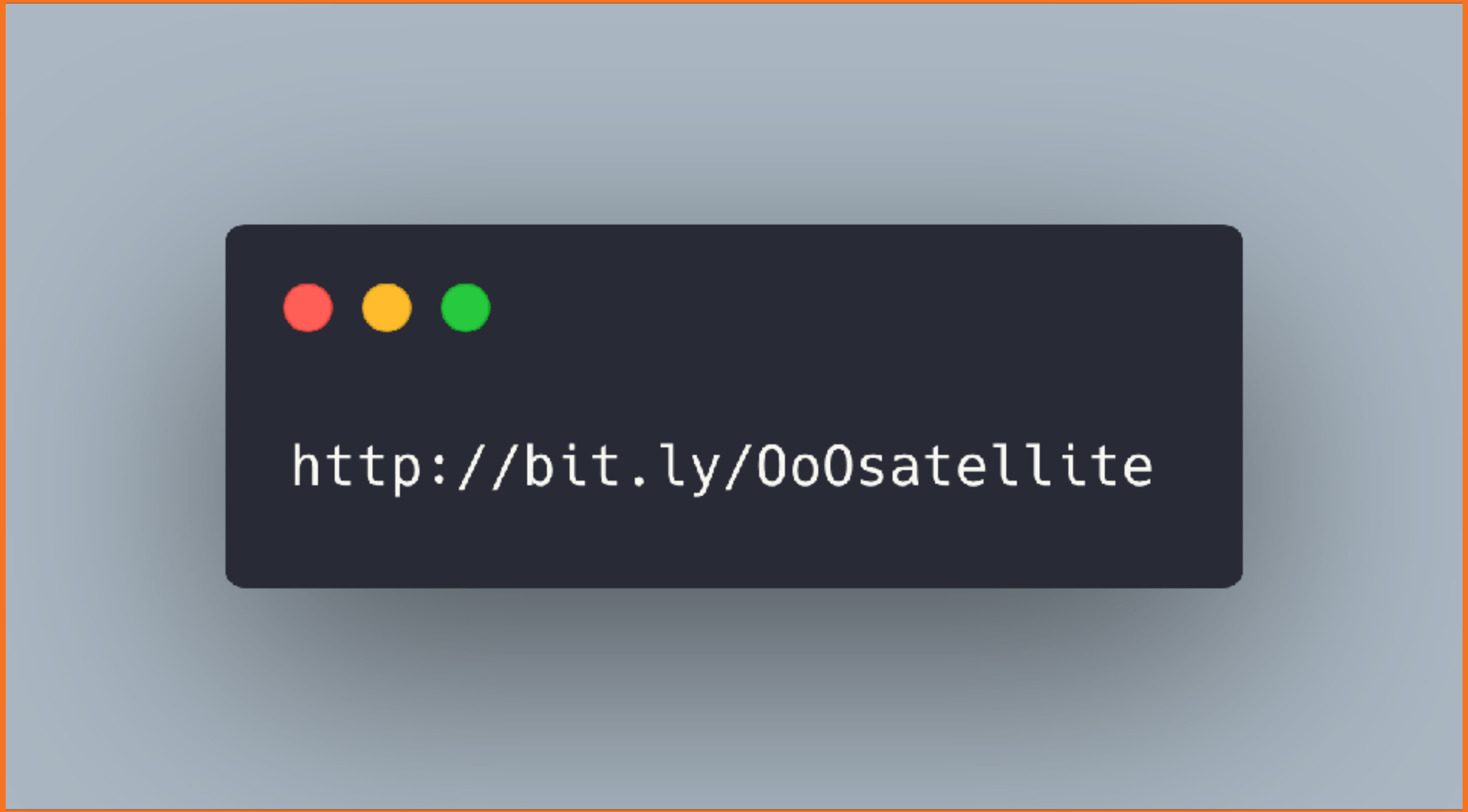




Lab time!

Let’s get to it!

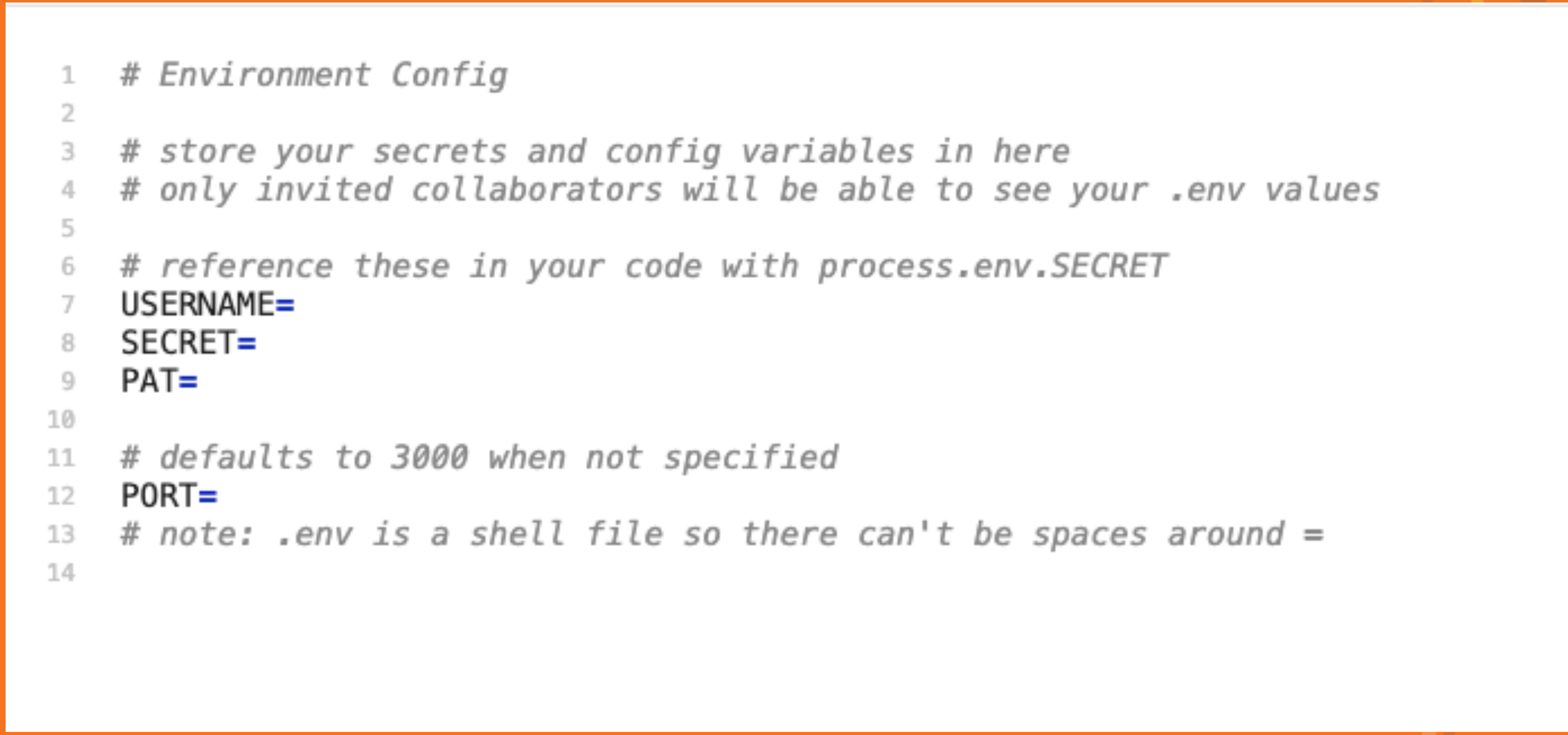
- Using glitch!



```
http://bit.ly/OoOsatellite
```

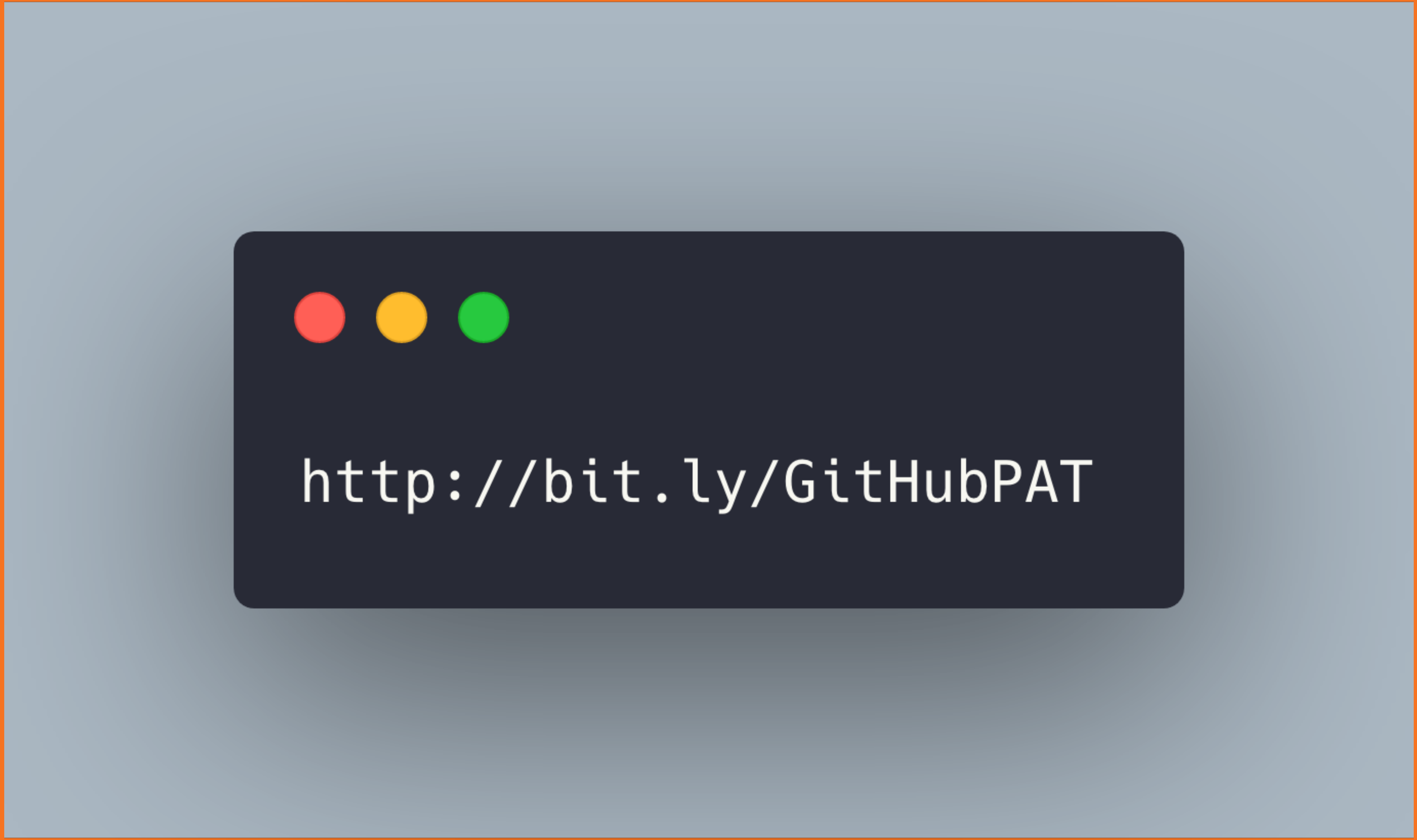
Look at the .env file

- USERNAME - Your GitHub username or org name
- SECRET - Webhook secret
- PAT - Personal access token



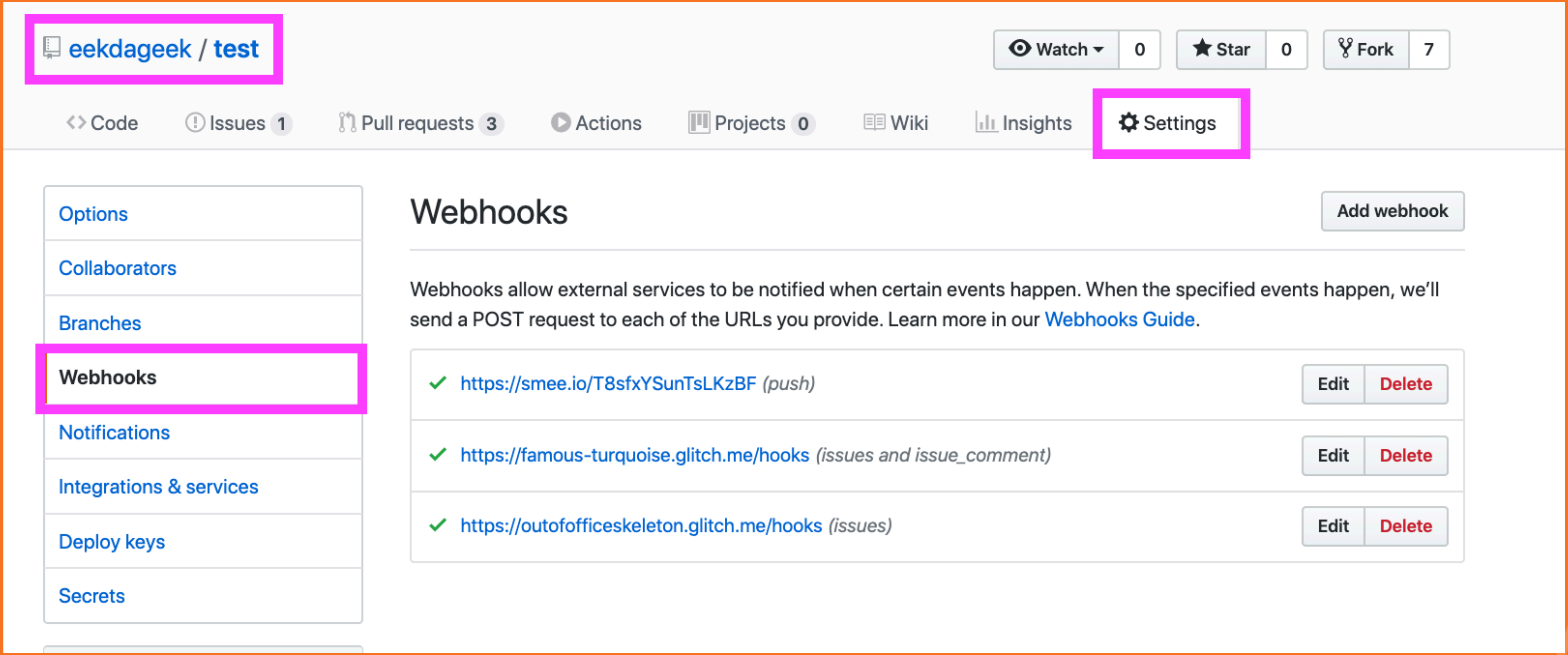
```
# Environment Config

# store your secrets and config variables in here
# only invited collaborators will be able to see your .env values

# reference these in your code with process.env.SECRET
USERNAME=
SECRET=
PAT=

# defaults to 3000 when not specified
PORT=
# note: .env is a shell file so there can't be spaces around =

```

Get a Personal Access Token



```
http://bit.ly/GitHubPAT
```

Set up webhook



eekdageek / test

Webhooks

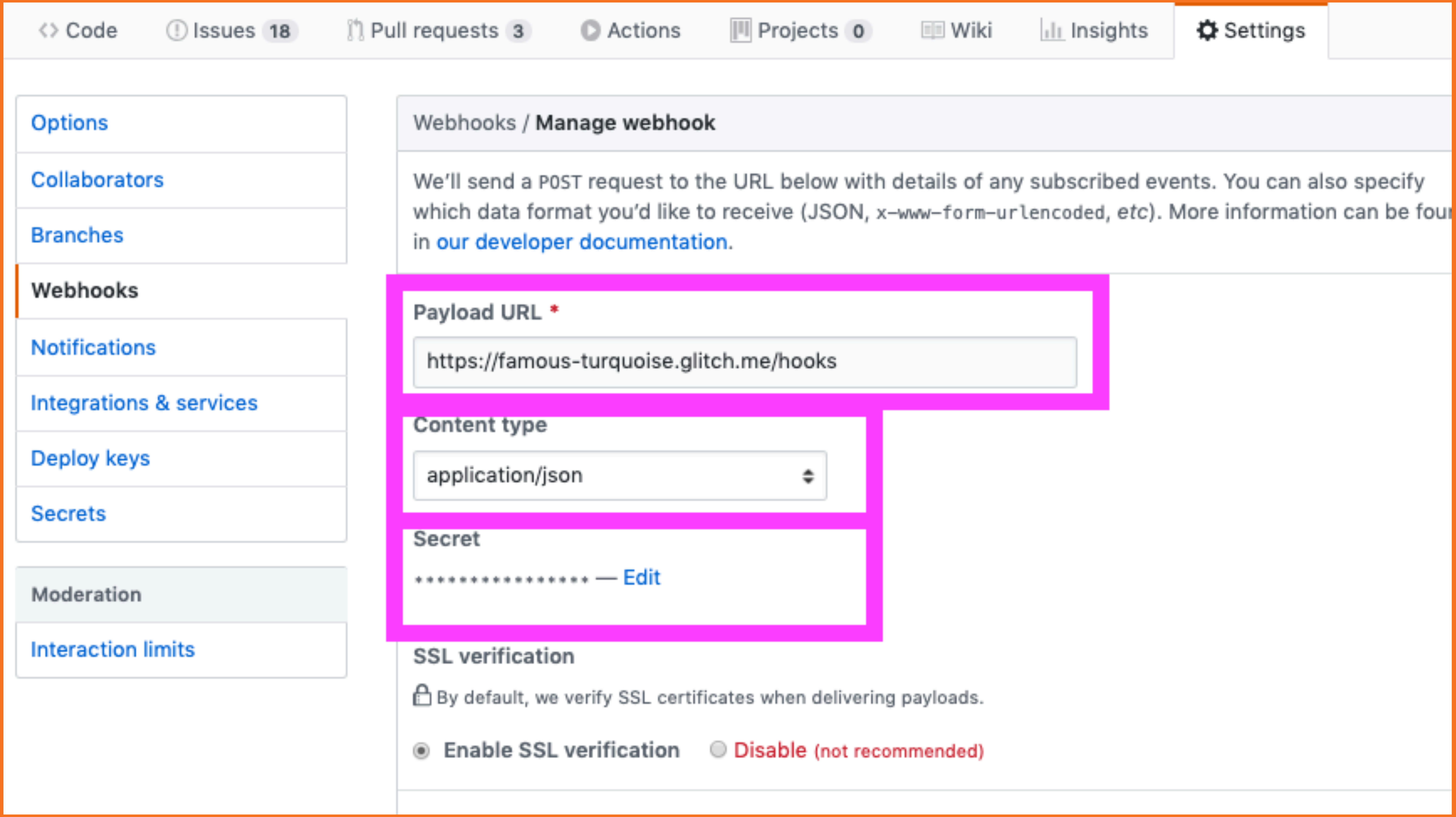
Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.

✓ https://smee.io/T8sfxYSunTsLKzBF *(push)*

✓ https://famous-turquoise.glitch.me/hooks *(issues and issue_comment)*

✓ https://outofofficeskeleton.glitch.me/hooks *(issues)*

Webhook



Webhooks / **Manage webhook**

We'll send a `POST` request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, `x-www-form-urlencoded`, *etc*). More information can be fou in our developer documentation.

Payload URL *

https://famous-turquoise.glitch.me/hooks

Content type

application/json

Secret

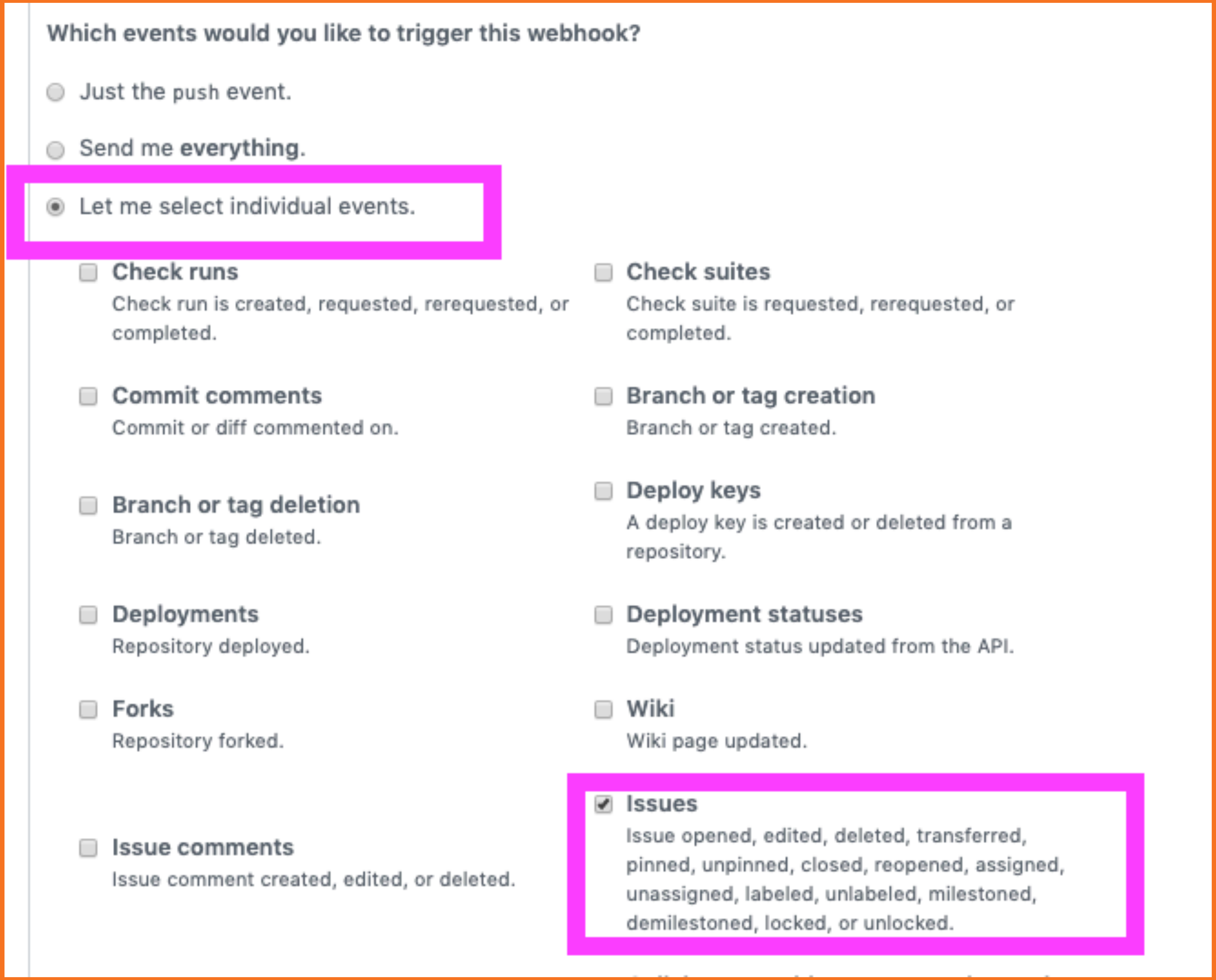
**************** — Edit

SSL verification

By default, we verify SSL certificates when delivering payloads.

[✓ Enable SSL verification] Disable (not recommended)

Webhook



Which events would you like to trigger this webhook?

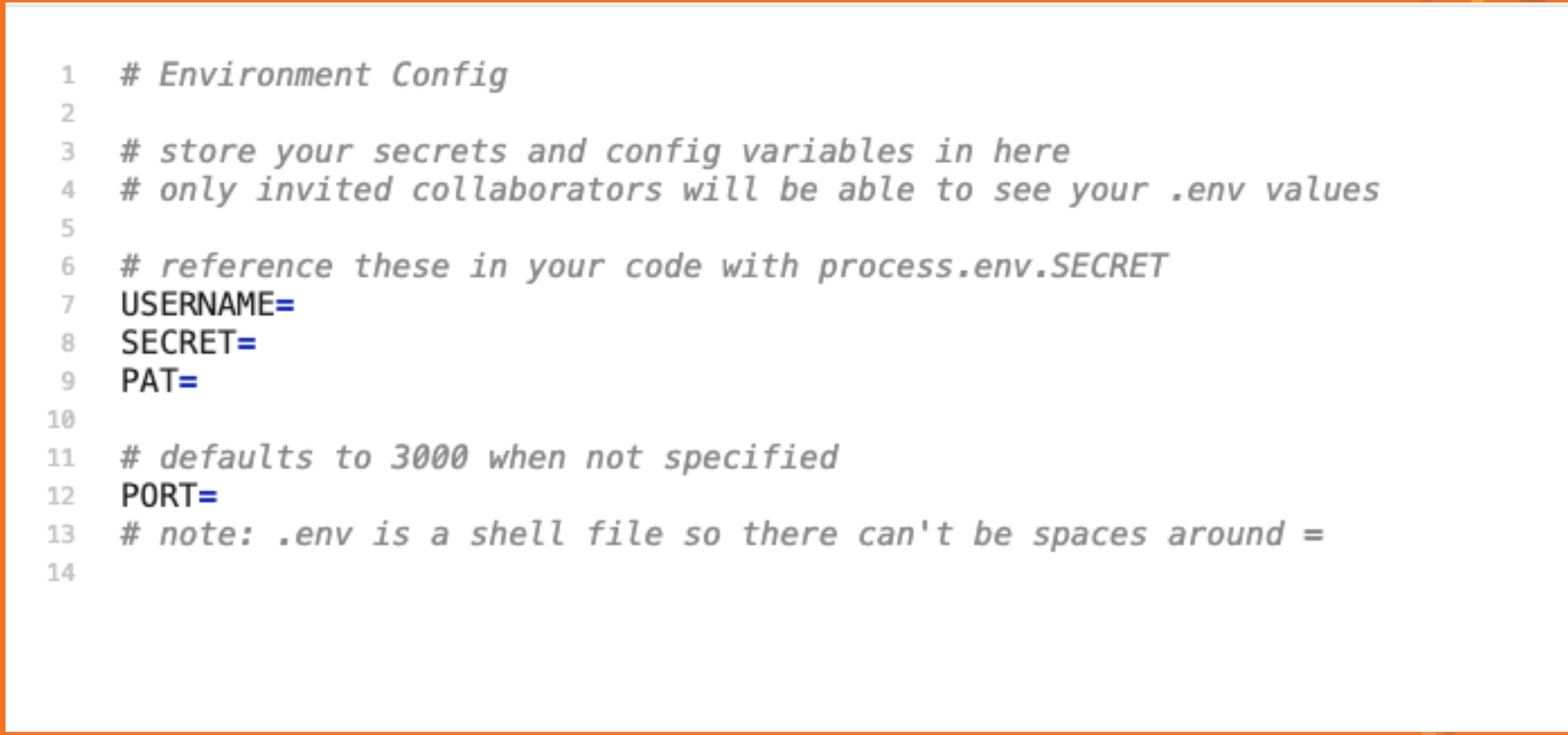
- ○ Just the push event.
- ○ Send me **everything**.
- ● Let me select individual events.

- ☐ **Check runs** Check run is created, requested, rerequested, or completed.
- ☐ **Check suites** Check suite is requested, rerequested, or completed.
- ☐ **Commit comments** Commit or diff commented on.
- ☐ **Branch or tag creation** Branch or tag created.
- ☐ **Branch or tag deletion** Branch or tag deleted.
- ☐ **Deploy keys** A deploy key is created or deleted from a repository.
- ☐ **Deployments** Repository deployed.
- ☐ **Deployment statuses** Deployment status updated from the API.
- ☐ **Forks** Repository forked.
- ☐ **Wiki** Wiki page updated.
- ☐ **Issue comments** Issue comment created, edited, or deleted.
- ☑ **Issues** Issue opened, edited, deleted, transferred, pinned, unpinned, closed, reopened, assigned, unassigned, labeled, unlabeled, milestoned, demilestoned, locked, or unlocked.

Let’s Go Back to Glitch!

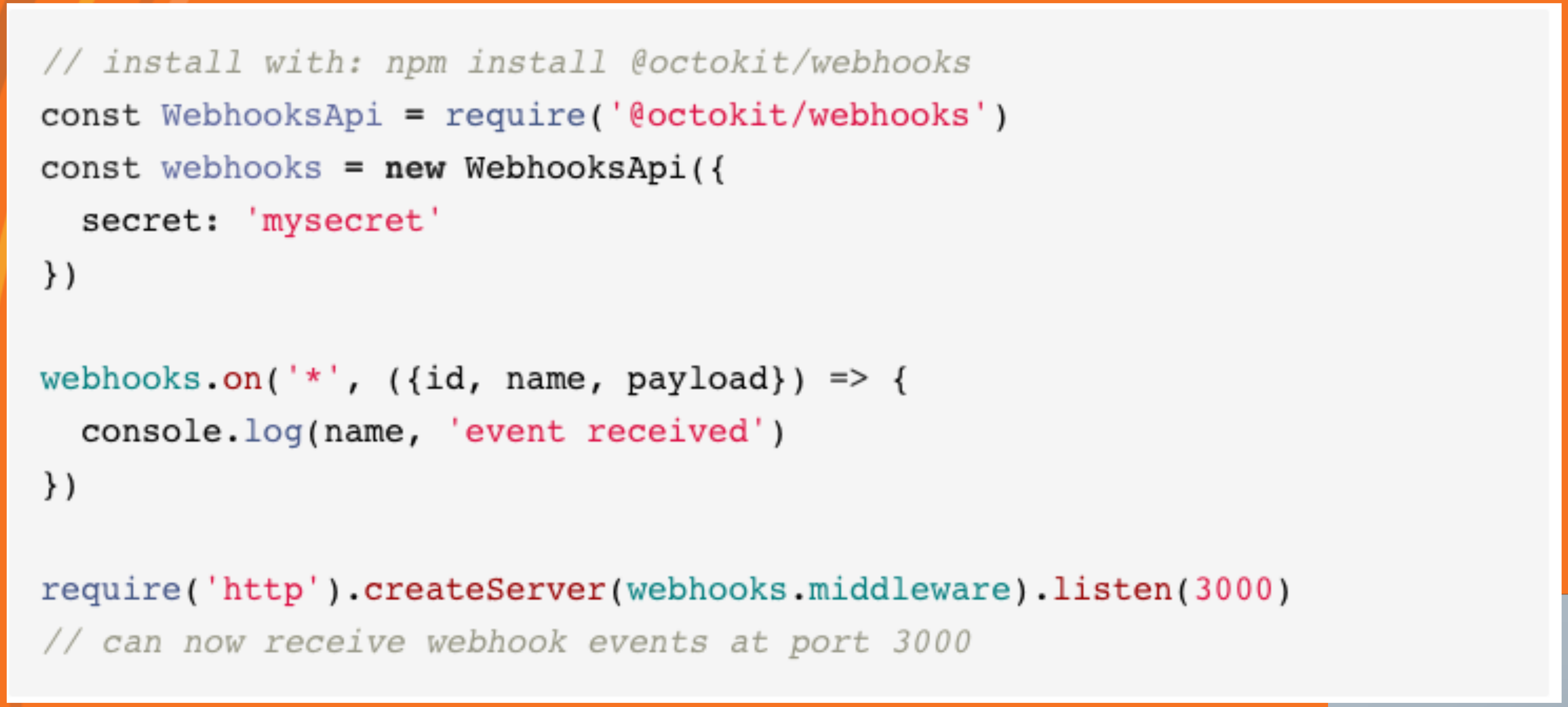
Edit the .env file

- USERNAME - Your GitHub username or org name
- SECRET - Webhook secret
- PAT - Personal access token

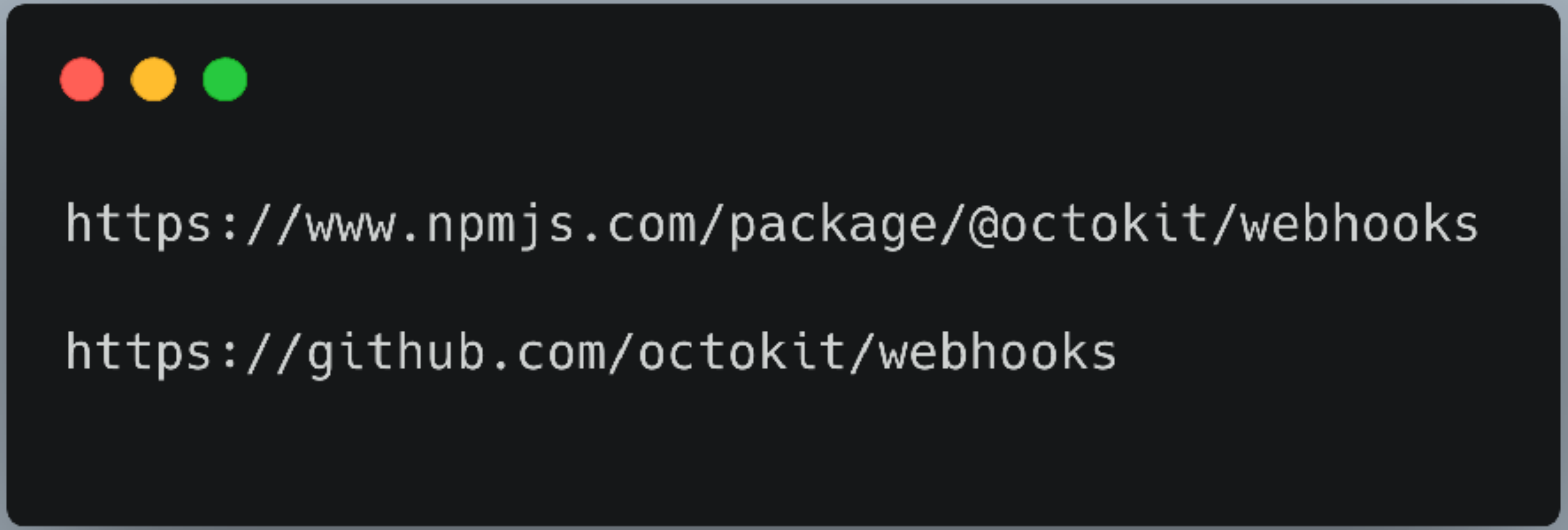


```
# Environment Config

# store your secrets and config variables in here
# only invited collaborators will be able to see your .env values

# reference these in your code with process.env.SECRET
USERNAME=
SECRET=
PAT=

# defaults to 3000 when not specified
PORT=
# note: .env is a shell file so there can't be spaces around =

```

@octokit/webhooks

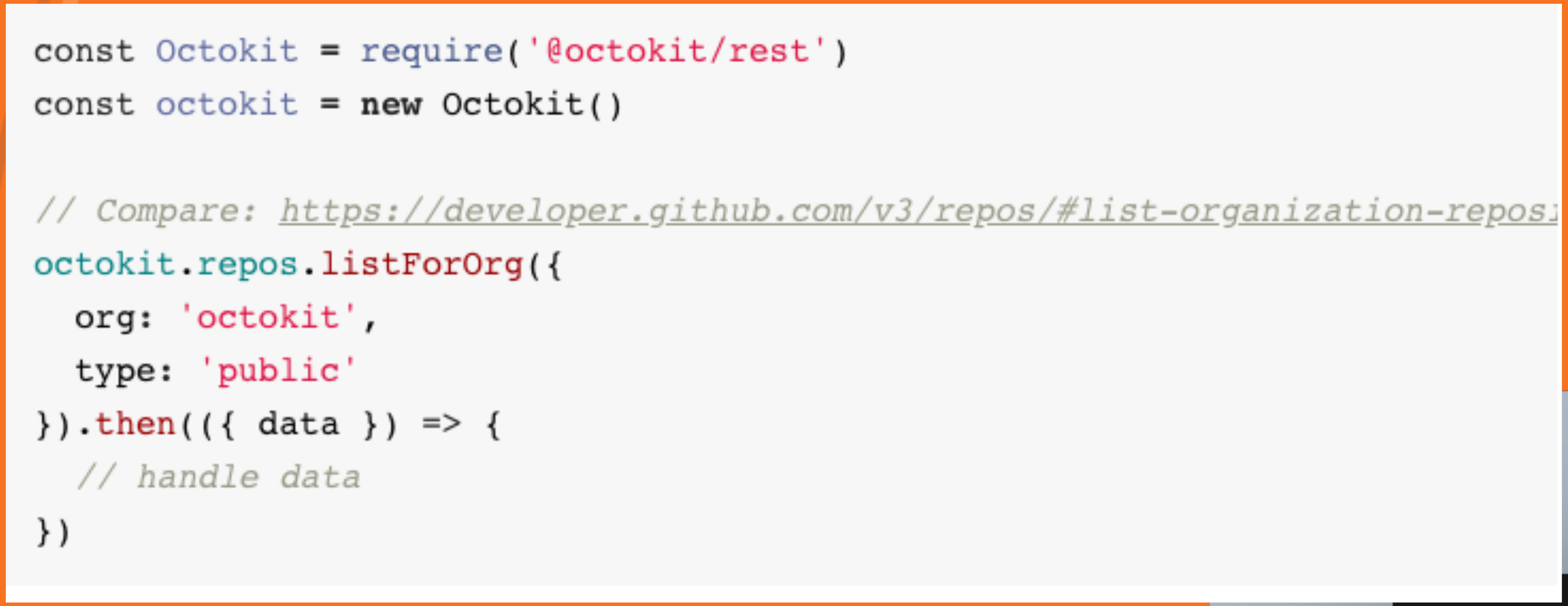


```
// install with: npm install @octokit/webhooks
const WebhooksApi = require('@octokit/webhooks')
const webhooks = new WebhooksApi({
  secret: 'mysecret'
})

webhooks.on('*', ({id, name, payload}) => {
  console.log(name, 'event received')
})

require('http').createServer(webhooks.middleware).listen(3000)
// can now receive webhook events at port 3000
```

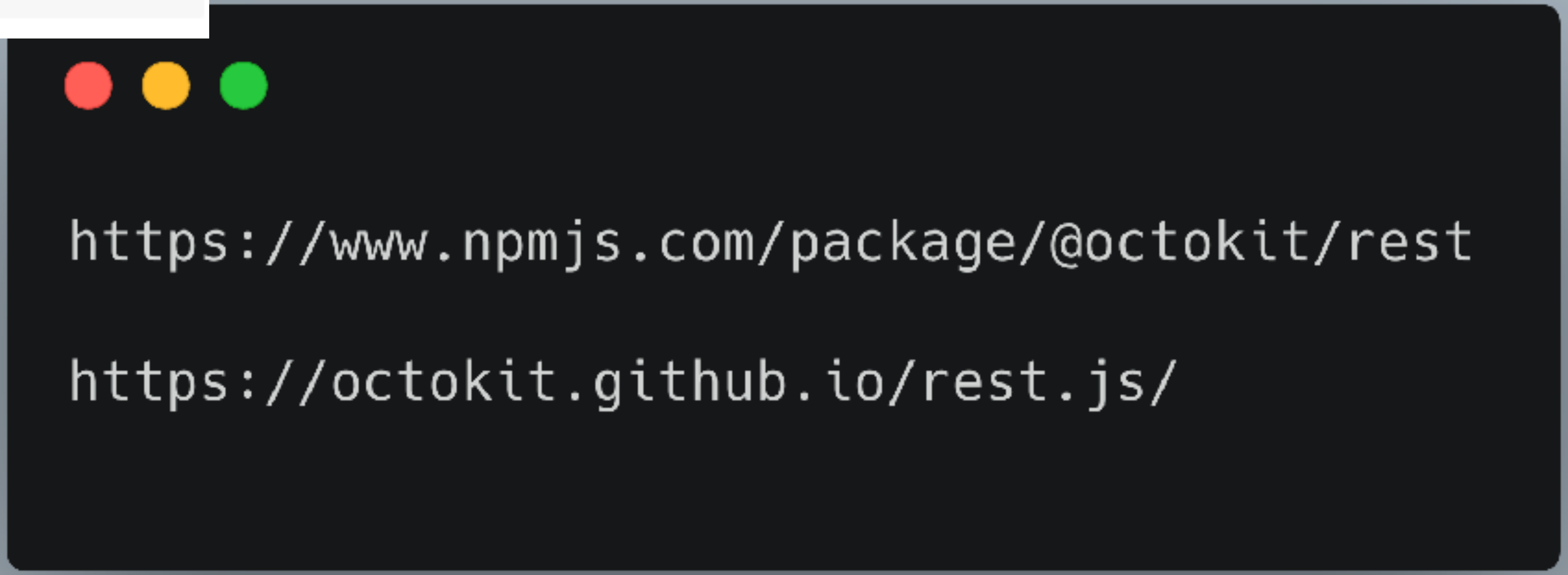


```
https://www.npmjs.com/package/@octokit/webhooks

https://github.com/octokit/webhooks
```

@octokit/rest

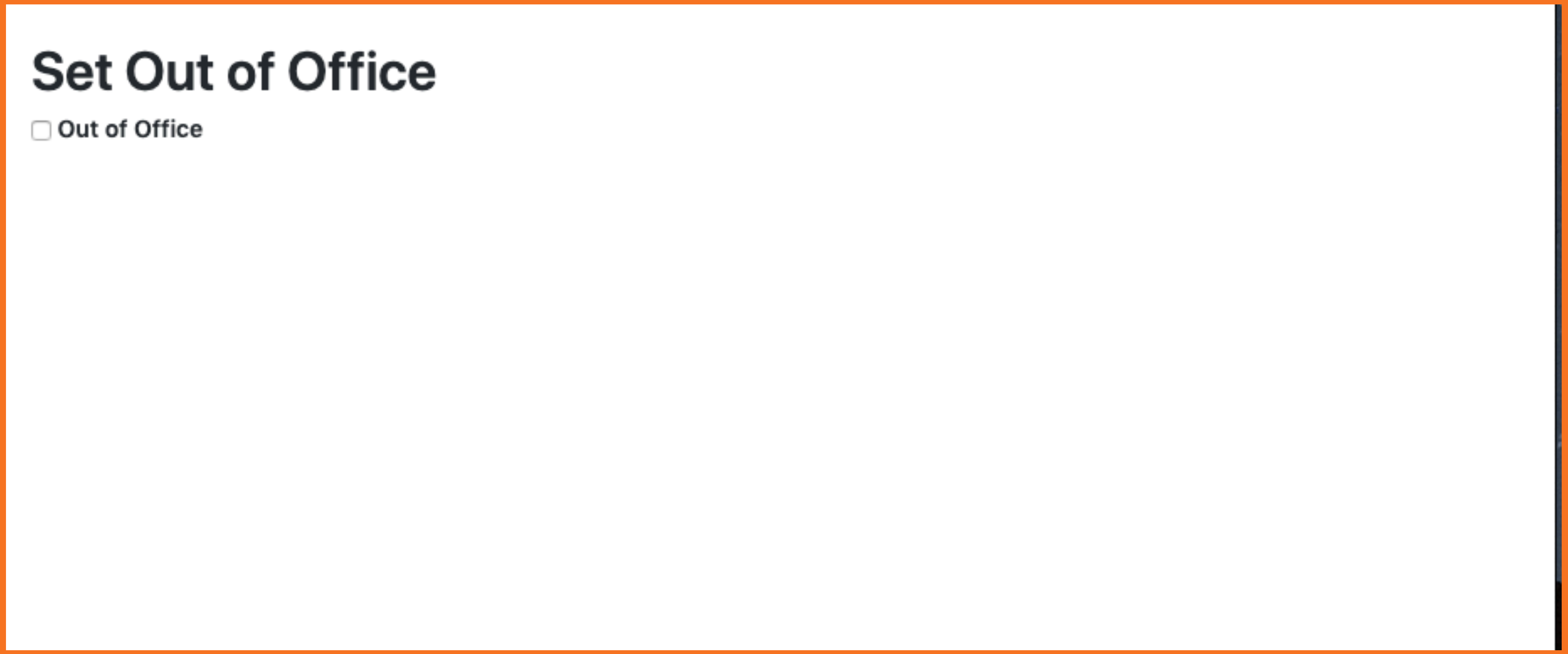


```
const Octokit = require('@octokit/rest')
const octokit = new Octokit()

// Compare: https://developer.github.com/v3/repos/#list-organization-repos
octokit.repos.listForOrg({
  org: 'octokit',
  type: 'public'
}).then(({ data }) => {
  // handle data
})
```



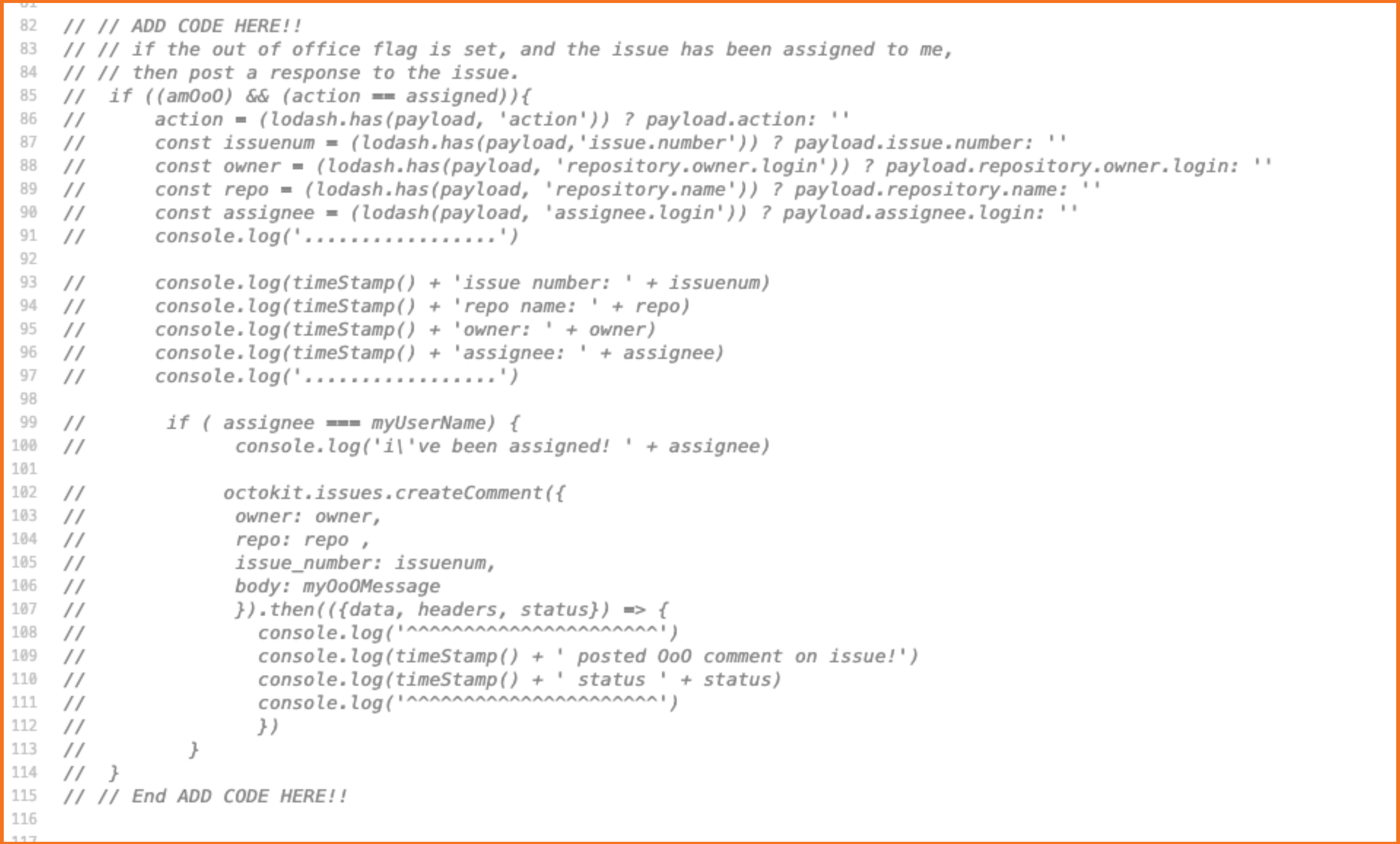
```
https://www.npmjs.com/package/@octokit/rest

https://octokit.github.io/rest.js/
```

Now click on the Out of Office checkbox


Set Out of Office

☐ Out of Office

Uncomment here in server.js



```
// // ADD CODE HERE!!
// // if the out of office flag is set, and the issue has been assigned to me,
// // then post a response to the issue.
//  if ((amOoO) && (action == assigned)){
//      action = (lodash.has(payload, 'action')) ? payload.action: ''
//      const issuenum = (lodash.has(payload,'issue.number')) ? payload.issue.number: ''
//      const owner = (lodash.has(payload, 'repository.owner.login')) ? payload.repository.owner.login: ''
//      const repo = (lodash.has(payload, 'repository.name')) ? payload.repository.name: ''
//      const assignee = (lodash(payload, 'assignee.login')) ? payload.assignee.login: ''
//      console.log('................')

//      console.log(timeStamp() + 'issue number: ' + issuenum)
//      console.log(timeStamp() + 'repo name: ' + repo)
//      console.log(timeStamp() + 'owner: ' + owner)
//      console.log(timeStamp() + 'assignee: ' + assignee)
//      console.log('................')

//       if ( assignee === myUserName) {
//             console.log('i\'ve been assigned! ' + assignee)

//            octokit.issues.createComment({
//             owner: owner,
//             repo: repo ,
//             issue_number: issuenum,
//             body: myOoOMessage
//             }).then(({data, headers, status}) => {
//               console.log('^^^^^^^^^^^^^^^^^^^^^^')
//               console.log(timeStamp() + ' posted OoO comment on issue!')
//               console.log(timeStamp() + ' status ' + status)
//               console.log('^^^^^^^^^^^^^^^^^^^^^^')
//               })
//         }
//  }
// // End ADD CODE HERE!!
```

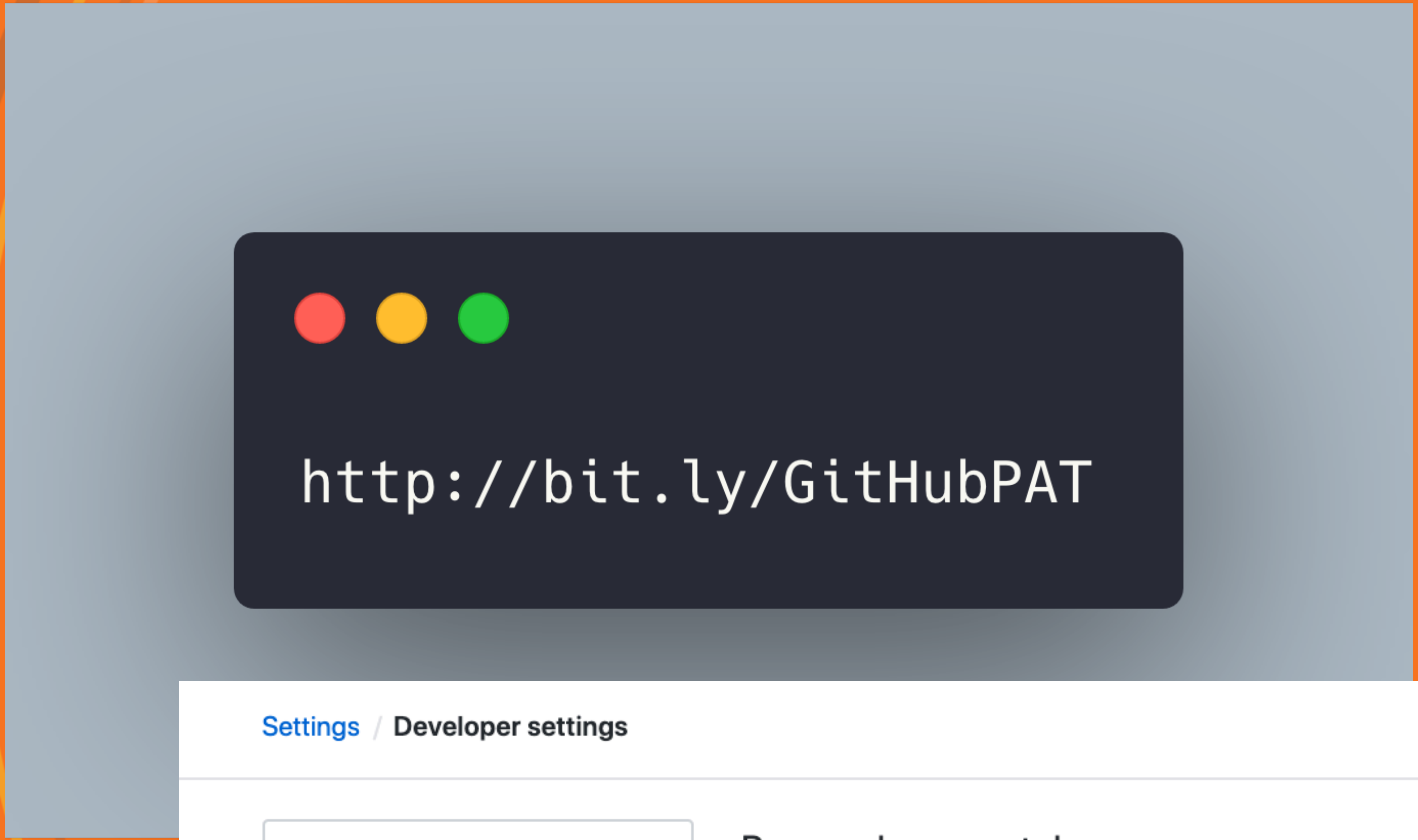
Neu-Venedig

BONUS

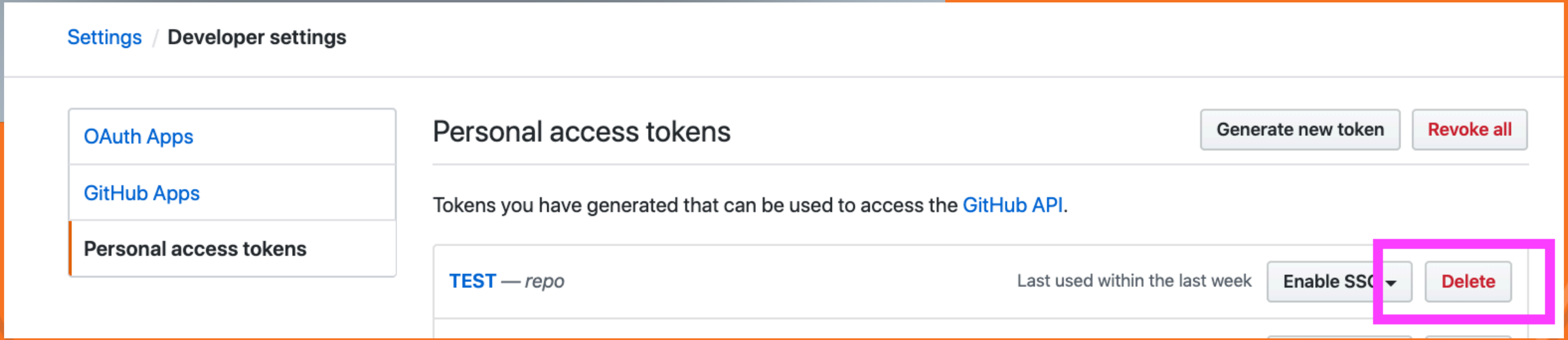
- If you assigned the issue yourself, don’t post the out of office response in the issue!

Tada!

Now, let's destroy that PAT!



```
http://bit.ly/GitHubPAT
```



Settings / Developer settings

Personal access tokens

Tokens you have generated that can be used to access the GitHub API.

TEST — *repo* Last used within the last week

Danke schön!

For questions, please email us at:

partnerengineering@github.com

SATELLITE

BERLIN 2019