Neverending Nightmare.js

Hi!

- working for Skype/Microsoft (Microsoft Teams)
- Zero Molecule (small agency)

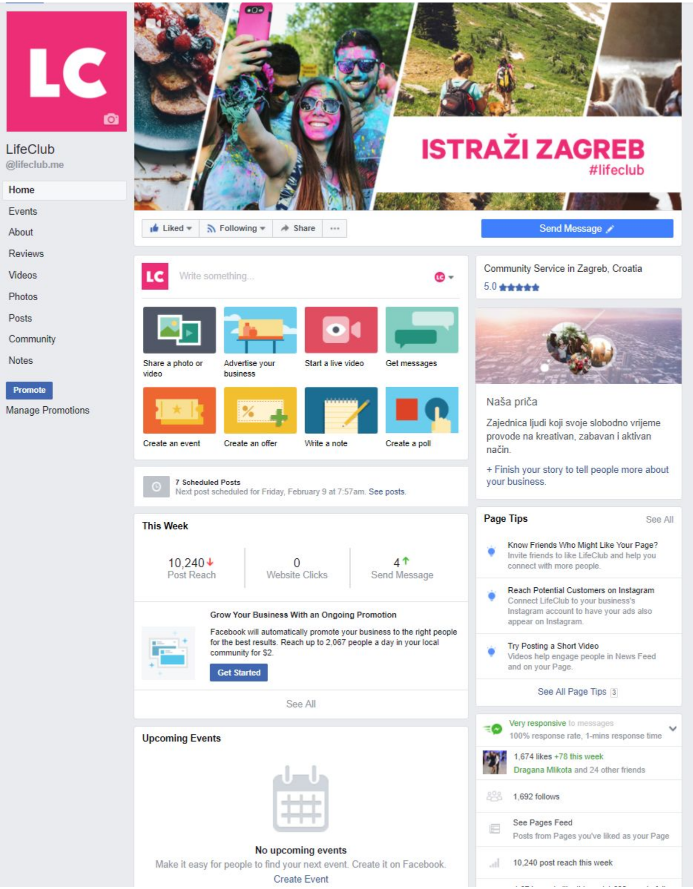
Intro

- automate browsing tasks for sites that don't have APIs
- wrapper around Electron

Nightmare.js

First try!

- Facebook page - **fb.com/lifeclub.me**
- Automated creation of **Notes**



Community Service in Zagreb, Croatia

5.0 ★★★★★

7 Scheduled Posts
Next post scheduled for Friday, February 9 at 7:57am. See posts.

10,240 Post Reach | 0 Website Clicks | 4 Send Message

Grow Your Business With an Ongoing Promotion

Facebook will automatically promote your business to the right people for the best results. Reach up to 2,067 people a day in your local community for $2.

No upcoming events

Make it easy for people to find your next event. Create it on Facebook.

Naša priča

Zajednica ljudi koji svoje slobodno vrijeme provode na kreativan, zabavan i aktivan način.

+ Finish your story to tell people more about your business.

Know Friends Who Might Like Your Page?
Invite friends to like LifeClub and help you connect with more people.

Reach Potential Customers on Instagram
Connect LifeClub to your business's Instagram account to have your ads also appear on Instagram.

Try Posting a Short Video
Videos help engage people in News Feed and on your Page.

Very responsive to messages
100% response rate, 1-mins response time

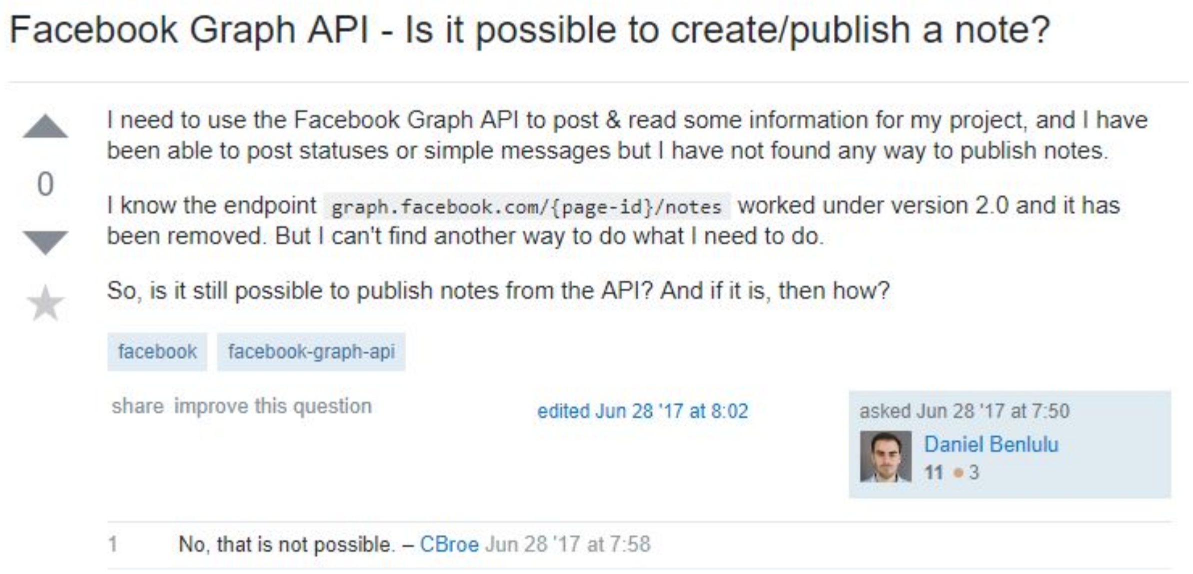
1,674 likes +78 this week
Dragana Mlikota and 24 other friends

1,692 follows

See Pages Feed
Posts from Pages you've liked as your Page

10,240 post reach this week

Why not Facebook API?



Facebook Graph API - Is it possible to create/publish a note?

I need to use the Facebook Graph API to post & read some information for my project, and I have been able to post statuses or simple messages but I have not found any way to publish notes.

I know the endpoint `graph.facebook.com/{page-id}/notes` worked under version 2.0 and it has been removed. But I can't find another way to do what I need to do.

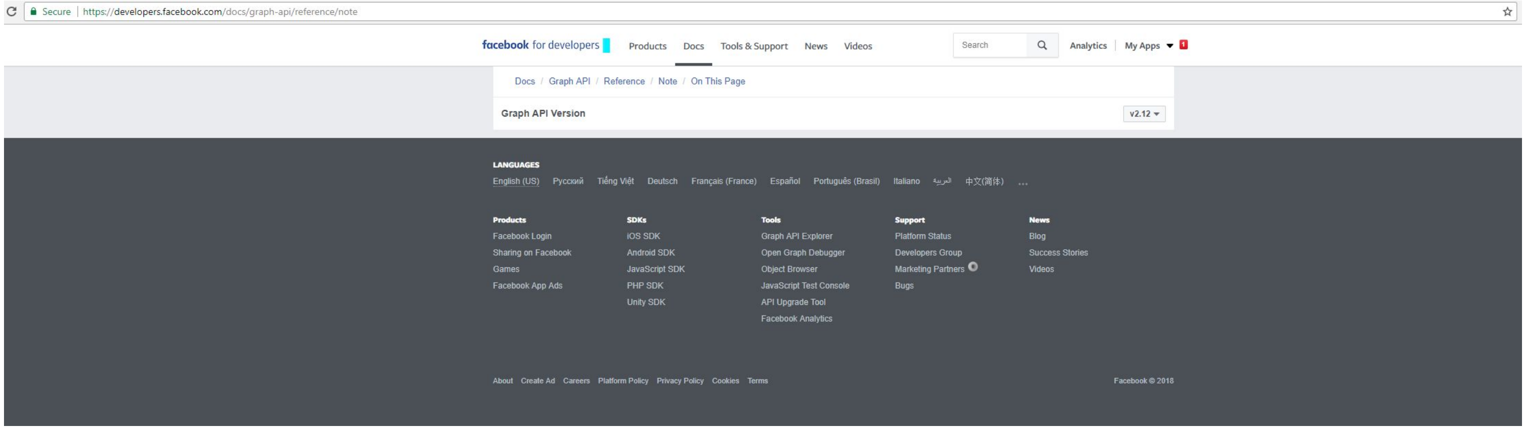
So, is it still possible to publish notes from the API? And if it is, then how?

edited Jun 28 '17 at 8:02

asked Jun 28 '17 at 7:50
Daniel Benlulu
11 ● 3

1 No, that is not possible. – CBroe Jun 28 '17 at 7:58

Why not Facebook API?



Secure | https://developers.facebook.com/docs/graph-api/reference/note

facebook for developers

Products Docs Tools & Support News Videos

Search

Analytics | My Apps

Docs / Graph API / Reference / Note / On This Page

Graph API Version v2.12

LANGUAGES

English (US) Русский Tiếng Việt Deutsch Français (France) Español Português (Brasil) Italiano العربية 中文(简体) ...

Products
Facebook Login
Sharing on Facebook
Games
Facebook App Ads

SDKs
iOS SDK
Android SDK
JavaScript SDK
PHP SDK
Unity SDK

Tools
Graph API Explorer
Open Graph Debugger
Object Browser
JavaScript Test Console
API Upgrade Tool
Facebook Analytics

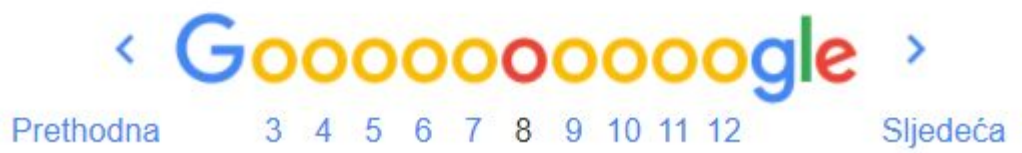
Support
Platform Status
Developers Group
Marketing Partners
Bugs

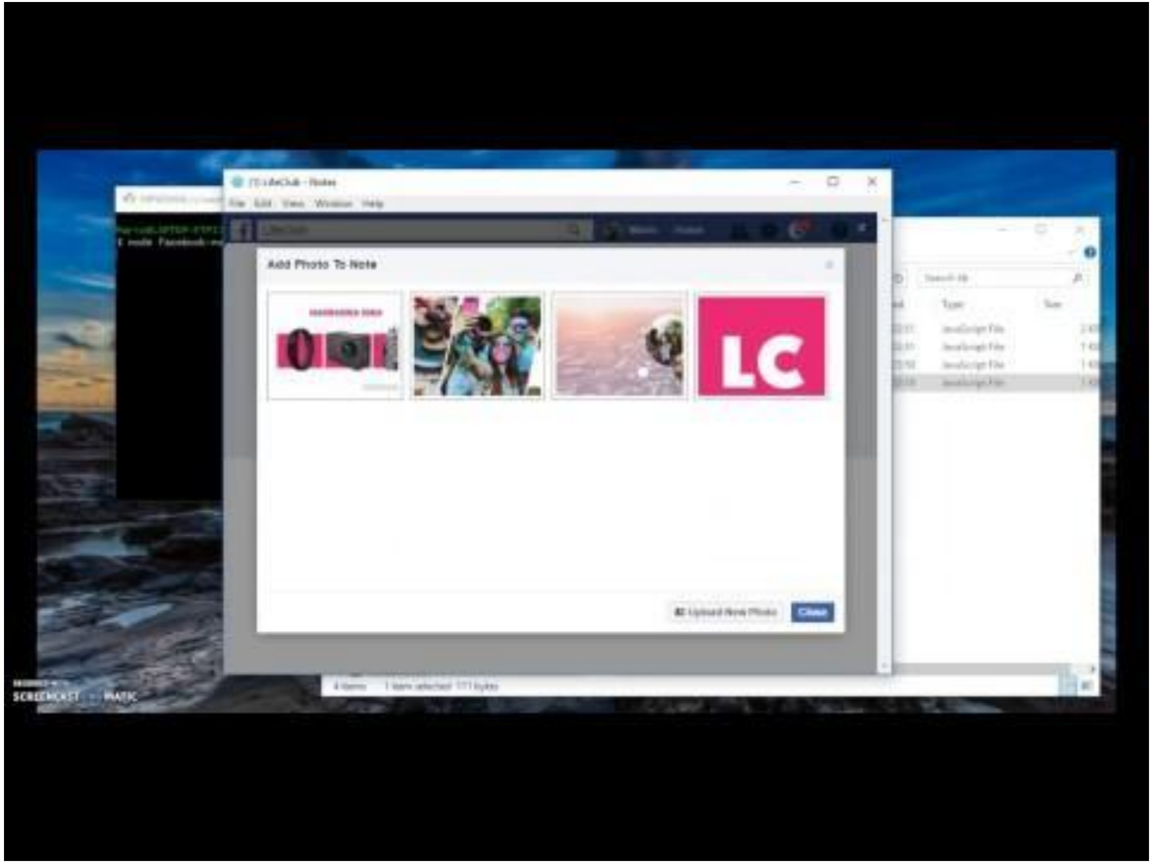
News
Blog
Success Stories
Videos

About Create Ad Careers Platform Policy Privacy Policy Cookies Terms

Facebook © 2018

Why not Facebook API?



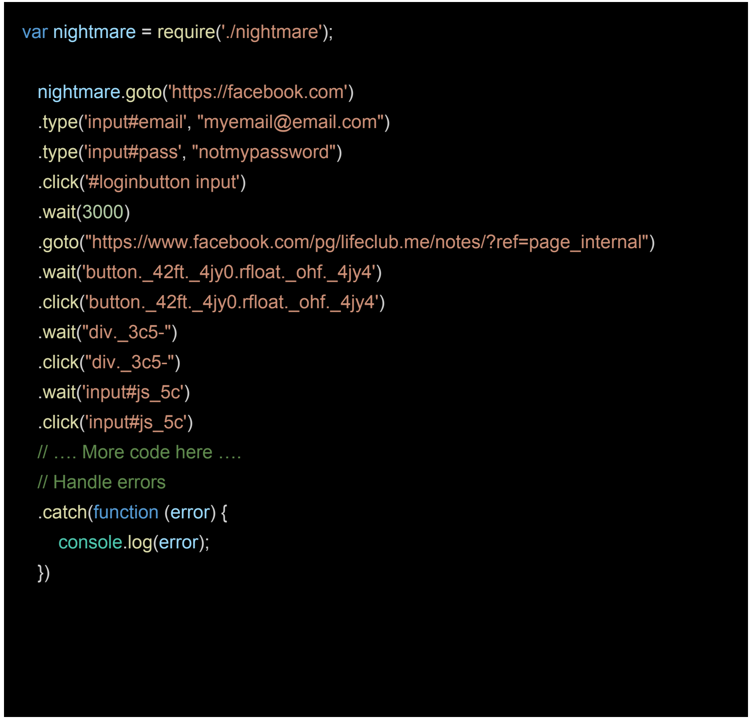

(1) LikeClub - Notes
File
Edit
View
Window
Help
Add Photo To Note
LC
Upload New Photo
Close
SCREENCAST-O-MATIC

Code

Really basic prototype

Stored FB notes in JSON

Automated tasks



```
var nightmare = require('./nightmare');

  nightmare.goto('https://facebook.com')
  .type('input#email', "myemail@email.com")
  .type('input#pass', "notmypassword")
  .click('#loginbutton input')
  .wait(3000)
  .goto("https://www.facebook.com/pg/lifeclub.me/notes/?ref=page_internal")
  .wait('button._42ft._4jy0.rfloat._ohf._4jy4')
  .click('button._42ft._4jy0.rfloat._ohf._4jy4')
  .wait("div._3c5-")
  .click("div._3c5-")
  .wait('input#js_5c')
  .click('input#js_5c')
  // …. More code here ….
  // Handle errors
  .catch(function (error) {
     console.log(error);
  })
```

Basic API

evaluate( )

Invokes callbacks. Used for DOM manipulation because code inside is browser scoped.

“listeners”

Similar to jQuery listener wrappers, there are “page” events

.on(“page”, function(type="confirm", message, response))

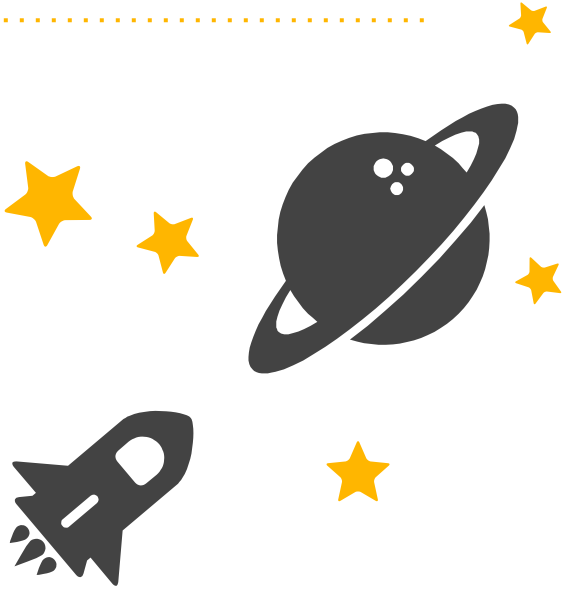
exists( )

Checks the existence of specific target selector

advanced

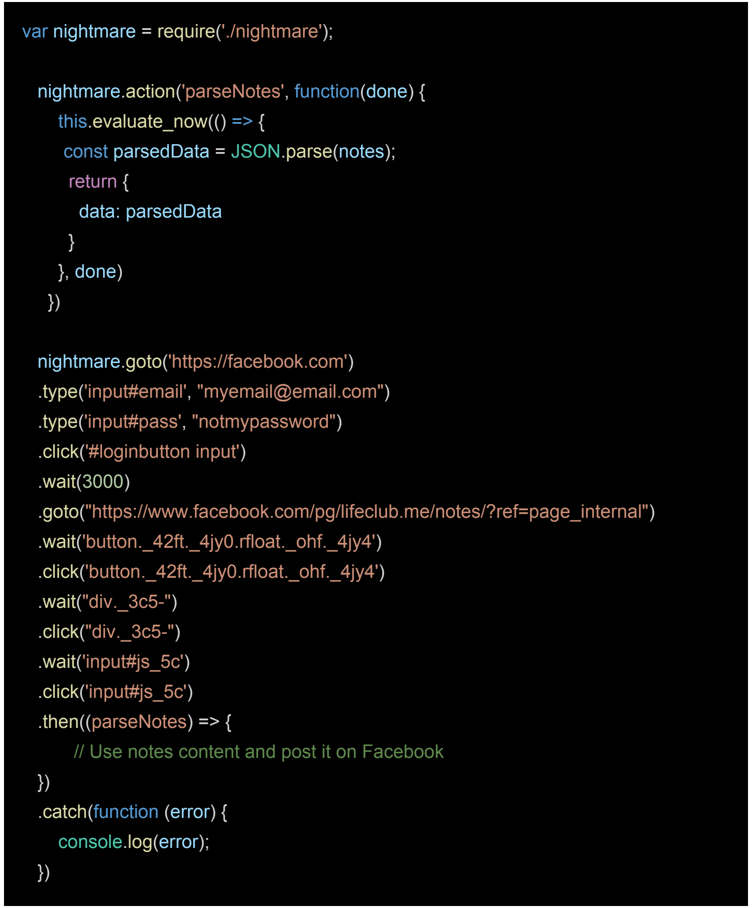
Promises, cookies, extensions etc

Extensions



- Adding custom functionality to Nightmare.js
- **Nightmare.action**

Code



```
var nightmare = require('./nightmare');

  nightmare.action('parseNotes', function(done) {
    this.evaluate_now(() => {
     const parsedData = JSON.parse(notes);
      return {
       data: parsedData
      }
    }, done)
   })

  nightmare.goto('https://facebook.com')
  .type('input#email', "myemail@email.com")
  .type('input#pass', "notmypassword")
  .click('#loginbutton input')
  .wait(3000)
  .goto("https://www.facebook.com/pg/lifeclub.me/notes/?ref=page_internal")
  .wait('button._42ft._4jy0.rfloat._ohf._4jy4')
  .click('button._42ft._4jy0.rfloat._ohf._4jy4')
  .wait("div._3c5-")
  .click("div._3c5-")
  .wait('input#js_5c')
  .click('input#js_5c')
  .then((parseNotes) => {
        // Use notes content and post it on Facebook
  })
  .catch(function (error) {
    console.log(error);
  })
```

Nightmare Plus extension

- Added support for iframes
- Supports waiting for user actions (clicks, checking etc)
- https://www.npmjs.com/package/nightmare-plus

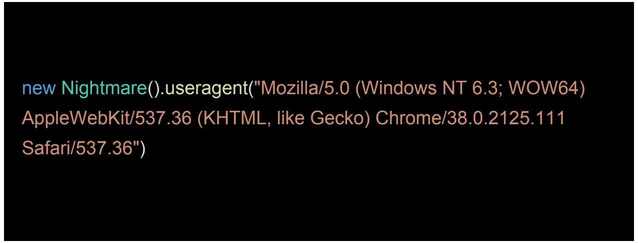
Daydream

- Very useful for recording actions
- Chrome extension
- https://github.com/segmentio/daydream

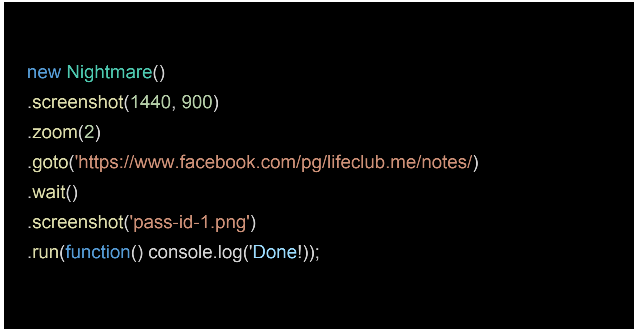
Testing (with) Nightmare

Testing env.

User Agents, Screenshots and Viewports



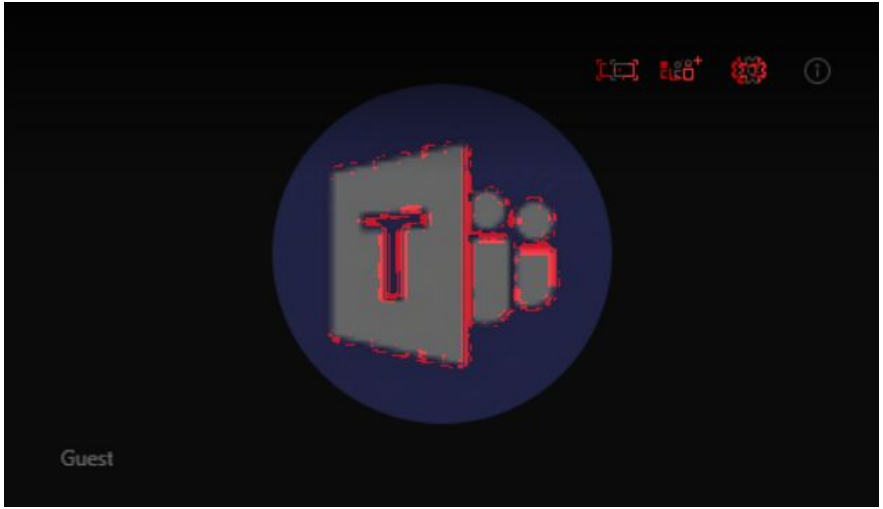
```
new Nightmare().useragent("Mozilla/5.0 (Windows NT 6.3; WOW64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/38.0.2125.111
Safari/537.36")
```



```
new Nightmare()
.screenshot(1440, 900)
.zoom(2)
.goto('https://www.facebook.com/pg/lifeclub.me/notes/)
.wait()
.screenshot('pass-id-1.png')
.run(function() console.log('Done!));
```

Automated visual testing

- Every PR that changes UI is tested against visual regressions
- Visual diffs are part of bigger build pipeline
- Works very good, but not perfect (needs “config” file as source of truth)

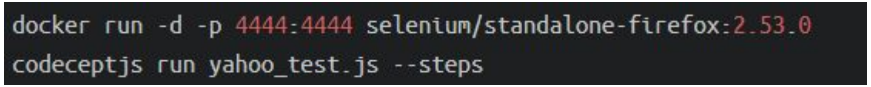


Niffy

- Diffing tool that can be integrated into Nightmare.js
- Scenario tests + visual diffs
- https://segment.com/blog/perceptual-diffing-with-niffy/

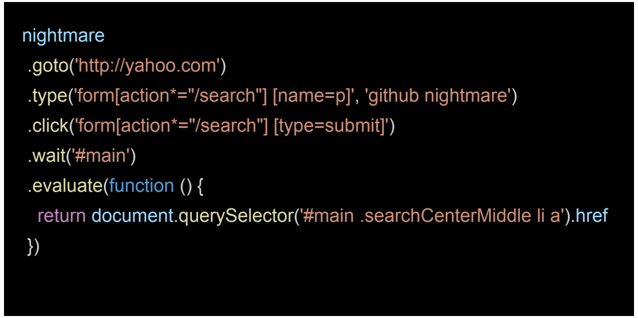
Performance

- It is expected for builds to last longer
- Guys at Codecept - Selenium vs Nightmare testing (more proof of concept)



```
docker run -d -p 4444:4444 selenium/standalone-firefox:2.53.0
codeceptjs run yahoo_test.js --steps
```

- Selenium - **17591** ms
- Nightmare - **5490** ms
- https://codecept.io/nightmare/



```
nightmare
 .goto('http://yahoo.com')
 .type('form[action*="/search"] [name=p]', 'github nightmare')
 .click('form[action*="/search"] [type=submit]')
 .wait('#main')
 .evaluate(function () {
   return document.querySelector('#main .searchCenterMiddle li a').href
 })
```

Conclusion

- Testing
- Building custom APIs
- Programming a repetitive task

Thanks!

Any questions?