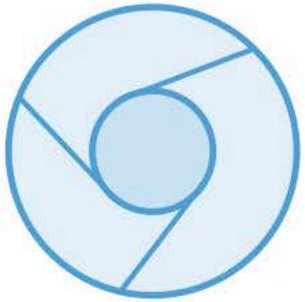
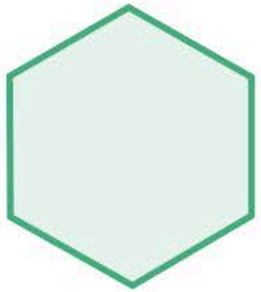
Make your Electron app feel at home everywhere

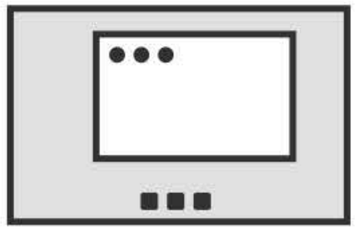
ELECTRON



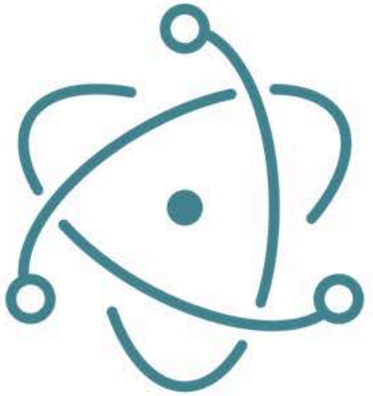
+



+



=



Chromium
for making
web pages

Node.js
for filesystems
and networks

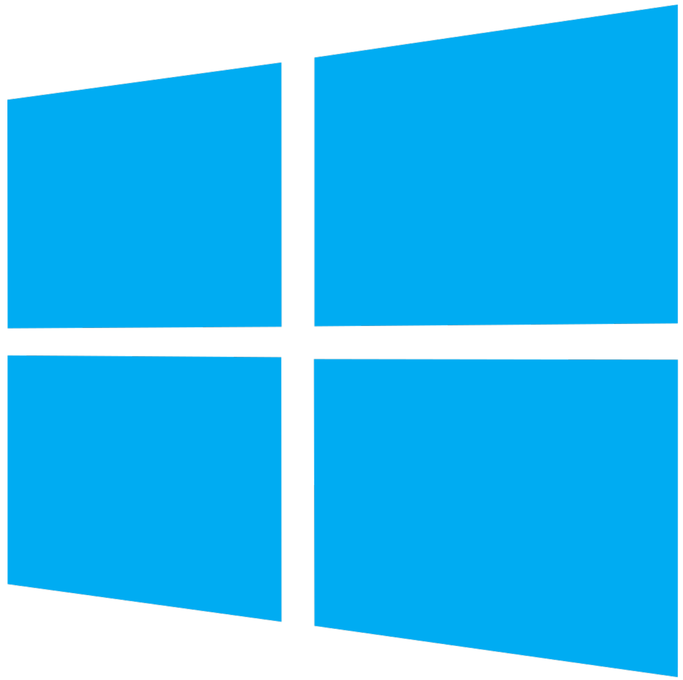
Native APIs
for three
systems

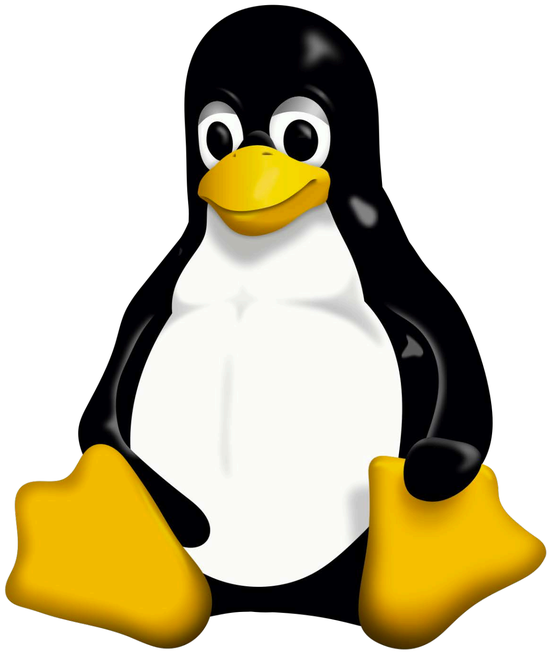
ELECTRON

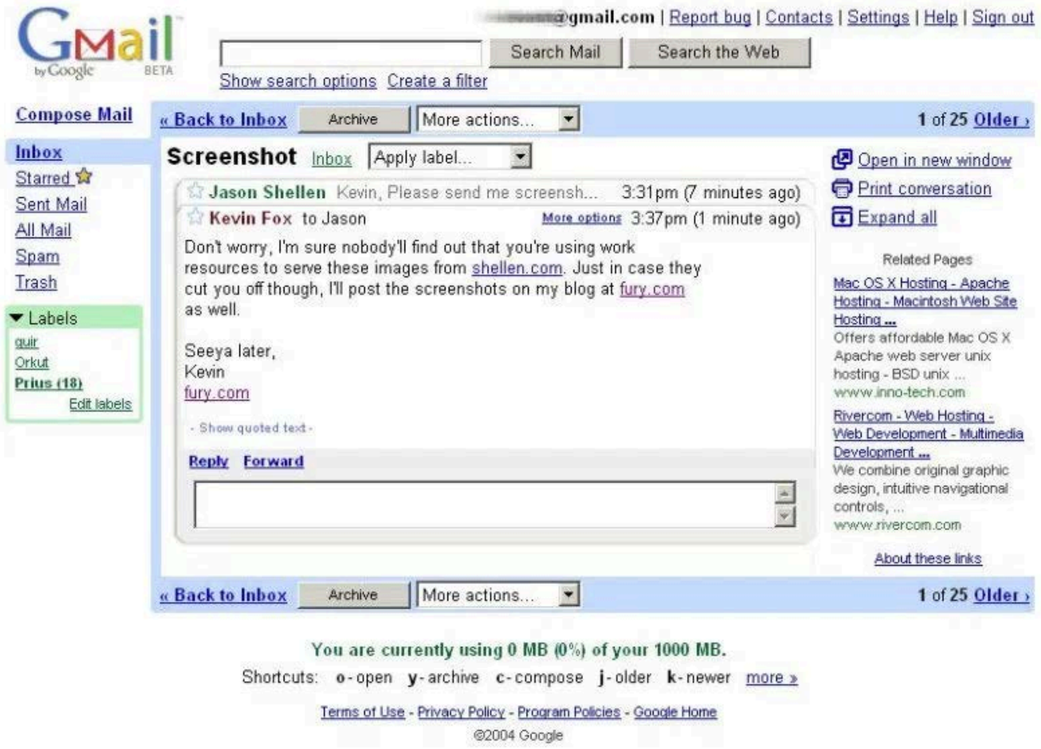
```
const { app, BrowserWindow } = require('electron');

app.whenReady().then(() => {
  let mainWindow = new BrowserWindow({
    width: 800,
    height: 600
  });

  mainWindow.loadURL('https://www.conf42.com/js2021');
})

app.on('window-all-closed', () {
  app.quit();
});
```

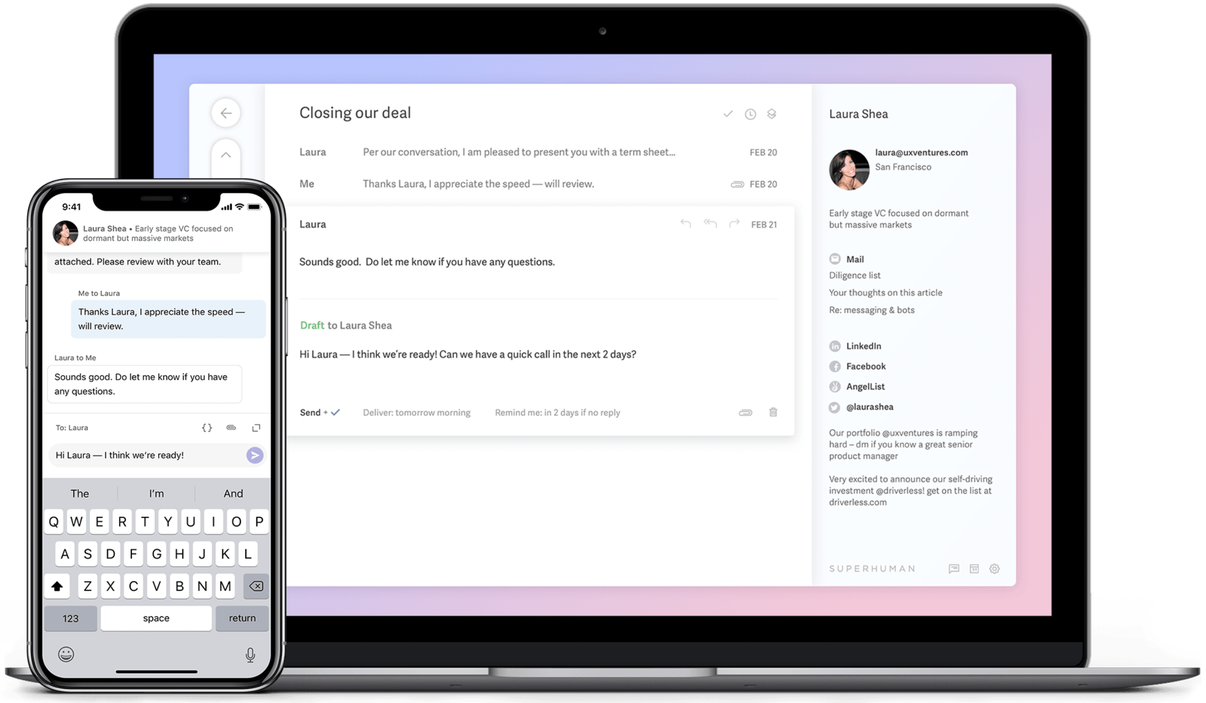
$ electron index.js

Mac OS





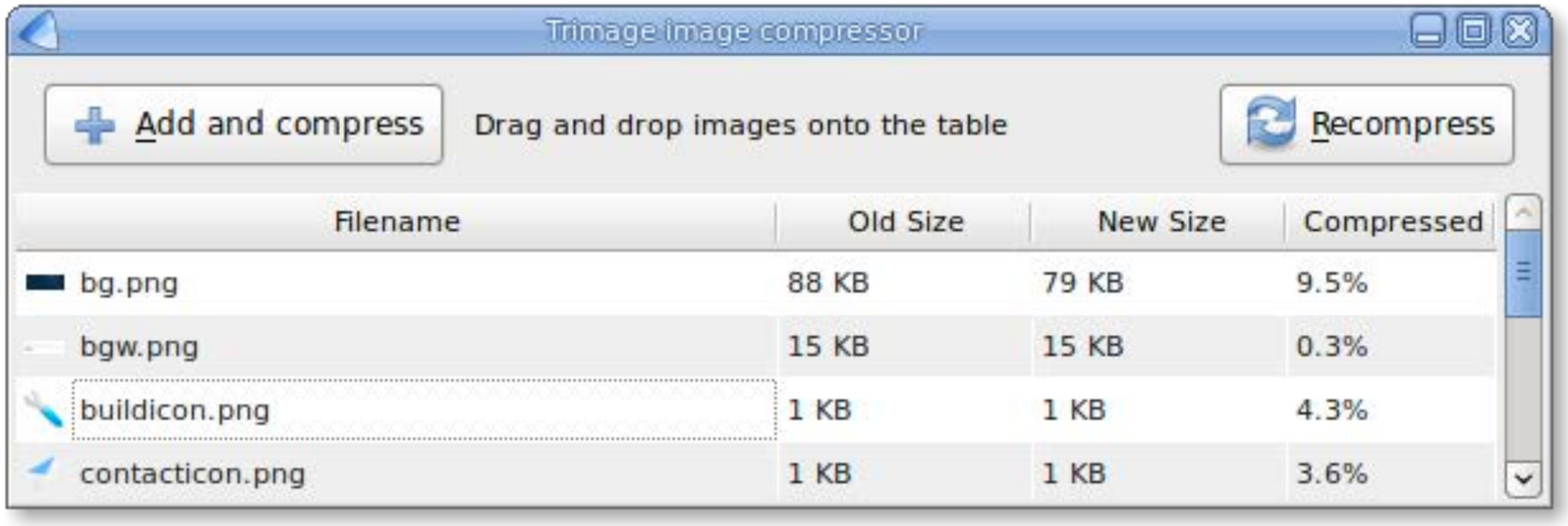

Gmail by Google BETA
@gmail.com | Report bug | Contacts | Settings | Help | Sign out
Search Mail
Search the Web
Show search options
Create a filter
Compose Mail
Inbox
Starred
Sent Mail
All Mail
Spam
Trash
Labels
quir
Orkut
Prius (18)
Edit labels
« Back to Inbox
Archive
More actions...
1 of 25 Older ›
Screenshot
Inbox
Apply label...
Jason Shellen Kevin, Please send me screensh... 3:31pm (7 minutes ago)
Kevin Fox to Jason
More options 3:37pm (1 minute ago)
Don't worry, I'm sure nobody'll find out that you're using work resources to serve these images from shellen.com. Just in case they cut you off though, I'll post the screenshots on my blog at fury.com as well.
Seeya later,
Kevin
fury.com
- Show quoted text -
Reply
Forward
Open in new window
Print conversation
Expand all
Related Pages
Mac OS X Hosting - Apache Hosting - Macintosh Web Site Hosting ...
Offers affordable Mac OS X Apache web server unix hosting - BSD unix ...
www.inno-tech.com
Rivercom - Web Hosting - Web Development - Multimedia Development ...
We combine original graphic design, intuitive navigational controls, ...
www.rivercom.com
About these links
« Back to Inbox
Archive
More actions...
1 of 25 Older ›
You are currently using 0 MB (0%) of your 1000 MB.
Shortcuts: o - open y - archive c - compose j - older k - newer more »
Terms of Use - Privacy Policy - Program Policies - Google Home
©2004 Google


Closing our deal
Laura
Per our conversation, I am pleased to present you with a term sheet...
FEB 20
Me
Thanks Laura, I appreciate the speed — will review.
FEB 20
Laura
FEB 21
Sounds good. Do let me know if you have any questions.
Draft to Laura Shea
Hi Laura — I think we're ready! Can we have a quick call in the next 2 days?
Send
Deliver: tomorrow morning
Remind me: in 2 days if no reply
Laura Shea
laura@uxventures.com
San Francisco
Early stage VC focused on dormant but massive markets
Mail
Diligence list
Your thoughts on this article
Re: messaging & bots
LinkedIn
Facebook
AngelList
@laurashea
Our portfolio @uxventures is ramping hard – dm if you know a great senior product manager
Very excited to announce our self-driving investment @driverless! get on the list at driverless.com
SUPERHUMAN
9:41
Laura Shea • Early stage VC focused on dormant but massive markets
attached. Please review with your team.
Me to Laura
Thanks Laura, I appreciate the speed — will review.
Laura to Me
Sounds good. Do let me know if you have any questions.
To: Laura
Hi Laura — I think we're ready!
The
I'm
And
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
123
space
return

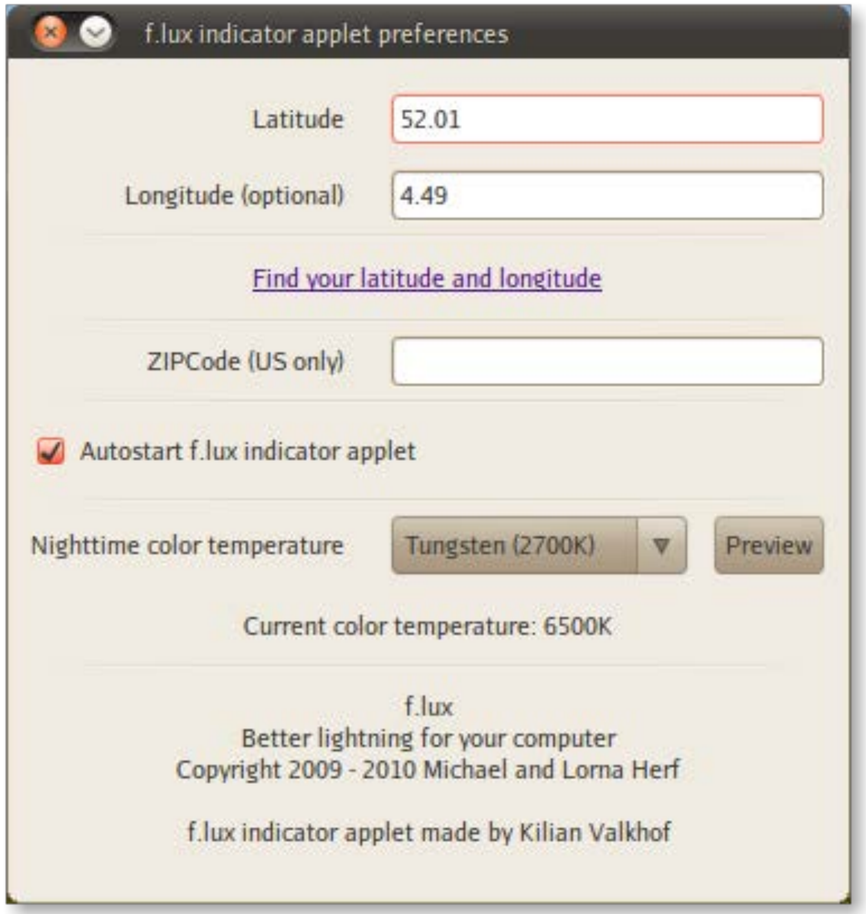
ELECTRON

Kilian Valkhof

kilianvalkhof.com | @kilianvalkhof

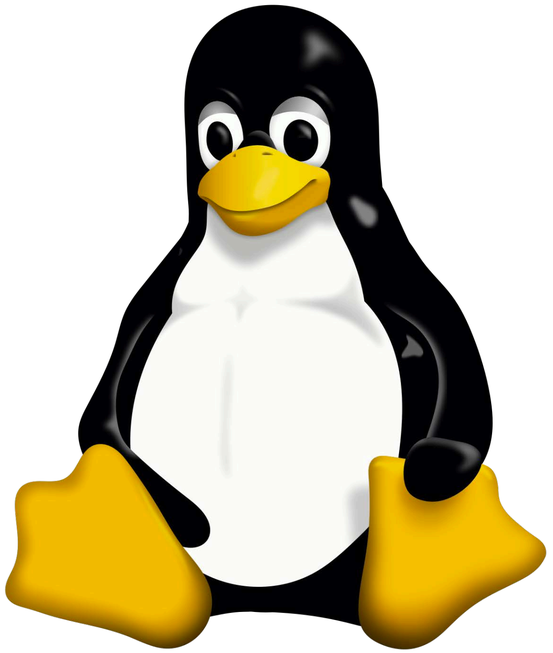

Trimage image compressor
Add and compress
Drag and drop images onto the table
Recompress
Filename
Old Size
New Size
Compressed
bg.png
88 KB
79 KB
9.5%
bgw.png
15 KB
15 KB
0.3%
buildicon.png
1 KB
1 KB
4.3%
contacticon.png
1 KB
1 KB
3.6%

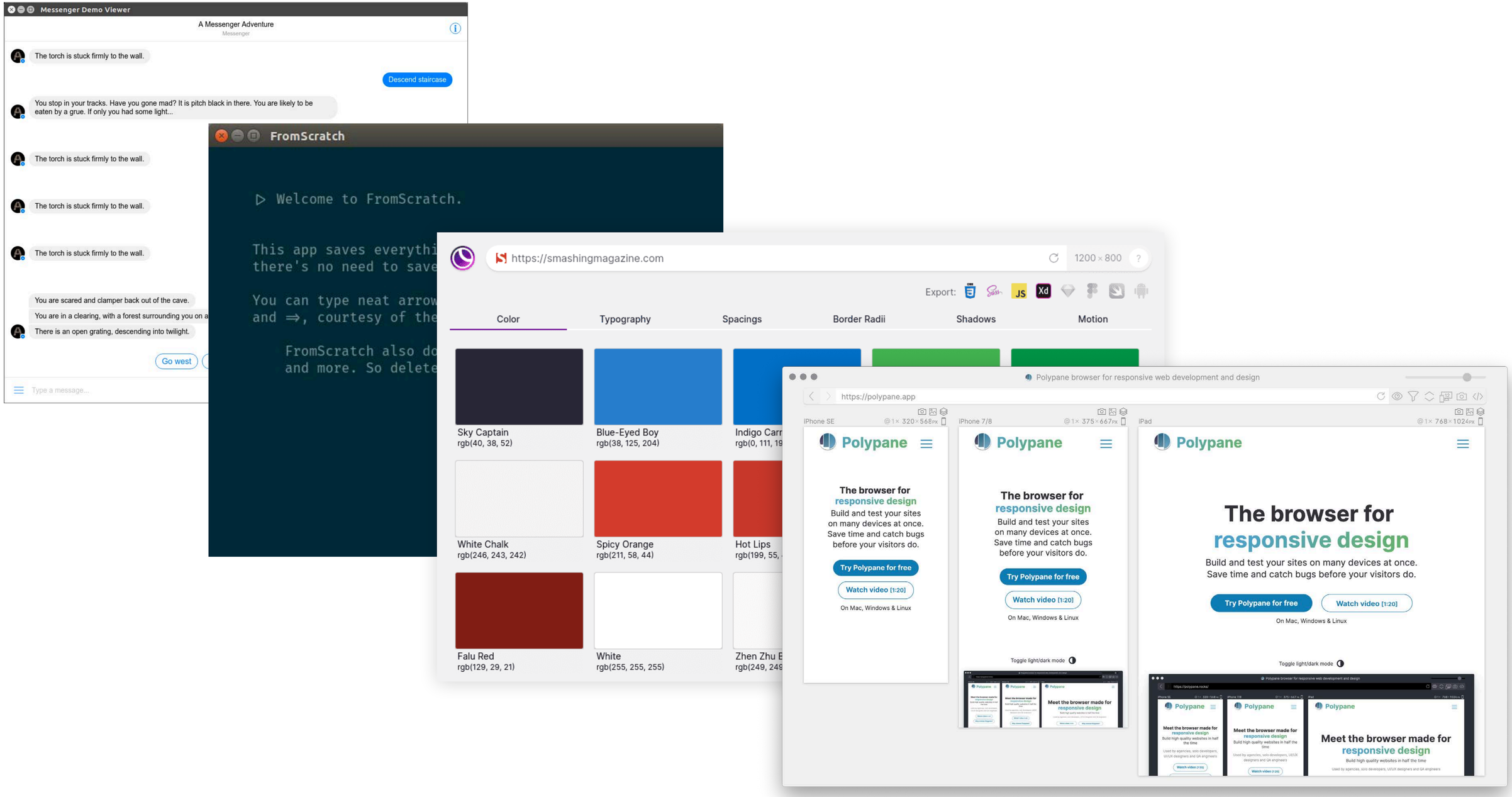
Qt


f.lux indicator applet preferences
Latitude
52.01
Longitude (optional)
4.49
Find your latitude and longitude
ZIPCode (US only)
Autostart f.lux indicator applet
Nighttime color temperature
Tungsten (2700K)
Preview
Current color temperature: 6500K
f.lux
Better lightning for your computer
Copyright 2009 - 2010 Michael and Lorna Herf
f.lux indicator applet made by Kilian Valkhof

GTK

Mac OS




Messenger Demo Viewer
A Messenger Adventure
Messenger
The torch is stuck firmly to the wall.
Descend staircase
You stop in your tracks. Have you gone mad? It is pitch black in there. You are likely to be eaten by a grue. If only you had some light...
The torch is stuck firmly to the wall.
The torch is stuck firmly to the wall.
The torch is stuck firmly to the wall.
You are scared and clamper back out of the cave.
You are in a clearing, with a forest surrounding you on a
There is an open grating, descending into twilight.
Go west
Type a message...
FromScratch
▷ Welcome to FromScratch.
This app saves everythi
there's no need to save
You can type neat arrow
and ⇒, courtesy of the
FromScratch also do
and more. So delete
https://smashingmagazine.com
1200 × 800
Export:
Color
Typography
Spacings
Border Radii
Shadows
Motion
Sky Captain
rgb(40, 38, 52)
Blue-Eyed Boy
rgb(38, 125, 204)
Indigo Carr
rgb(0, 111, 19
White Chalk
rgb(246, 243, 242)
Spicy Orange
rgb(211, 58, 44)
Hot Lips
rgb(199, 55,
Falu Red
rgb(129, 29, 21)
White
rgb(255, 255, 255)
Zhen Zhu B
rgb(249, 249
Polypane browser for responsive web development and design
https://polypane.app
iPhone SE
@1× 320×568px
iPhone 7/8
@1× 375×667px
iPad
@1× 768×1024px
Polypane
The browser for
responsive design
Build and test your sites on many devices at once. Save time and catch bugs before your visitors do.
Try Polypane for free
Watch video [1:20]
On Mac, Windows & Linux
Toggle light/dark mode
Meet the browser made for responsive design
Build high quality websites in half the time

8 ways to make your Electron app feel great on all three platforms

1.
Opening your app

```
const { app, BrowserWindow } = require('electron');

app.whenReady().then(() => {
  let mainWindow = new BrowserWindow({
    width: 800,
    height: 600
  });

  mainWindow.loadURL('https://www.conf42.com/js2021');
})

app.on('window-all-closed', () {
  app.quit();
});
```

$ electron index.js

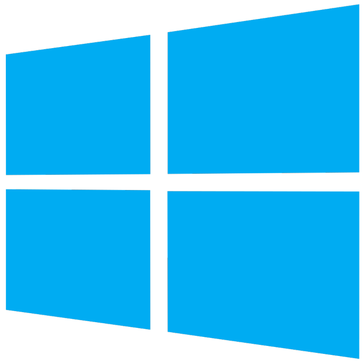
```
const { app, BrowserWindow } = require('electron');

app.whenReady().then(() => {
  let mainWindow = new BrowserWindow({
    width: 800,
    height: 600,
    show: false
  });

  mainWindow.once('ready-to-show', () => {
    mainWindow.show();
    mainWindow.focus();
  })

  mainWindow.loadURL('https://www.conf42.com/js2021');
})
```

Hide the app until the page has loaded

```
const { app, BrowserWindow } = require('electron');

app.whenReady().then(() => {
  let mainWindow = new BrowserWindow({
    width: 800,
    height: 600,
    show: false,

    backgroundColor: "#fafafa",
  });

  mainWindow.once('ready-to-show', () => {
    mainWindow.show();
    mainWindow.focus();
  })

  mainWindow.loadURL('https://www.conf42.com/js2021');
})
```

Use a background color that matches your app

2.
Closing your app



The app *has* windows

The window *is* the app

```
app.on('window-all-closed', () ⇒ {
  app.quit()
})
```

Windows and Linux:

All app windows closed = close the app

```
app.on('window-all-closed', () => {
  if (process.platform !== 'darwin') {
    app.quit()
  }
})
```

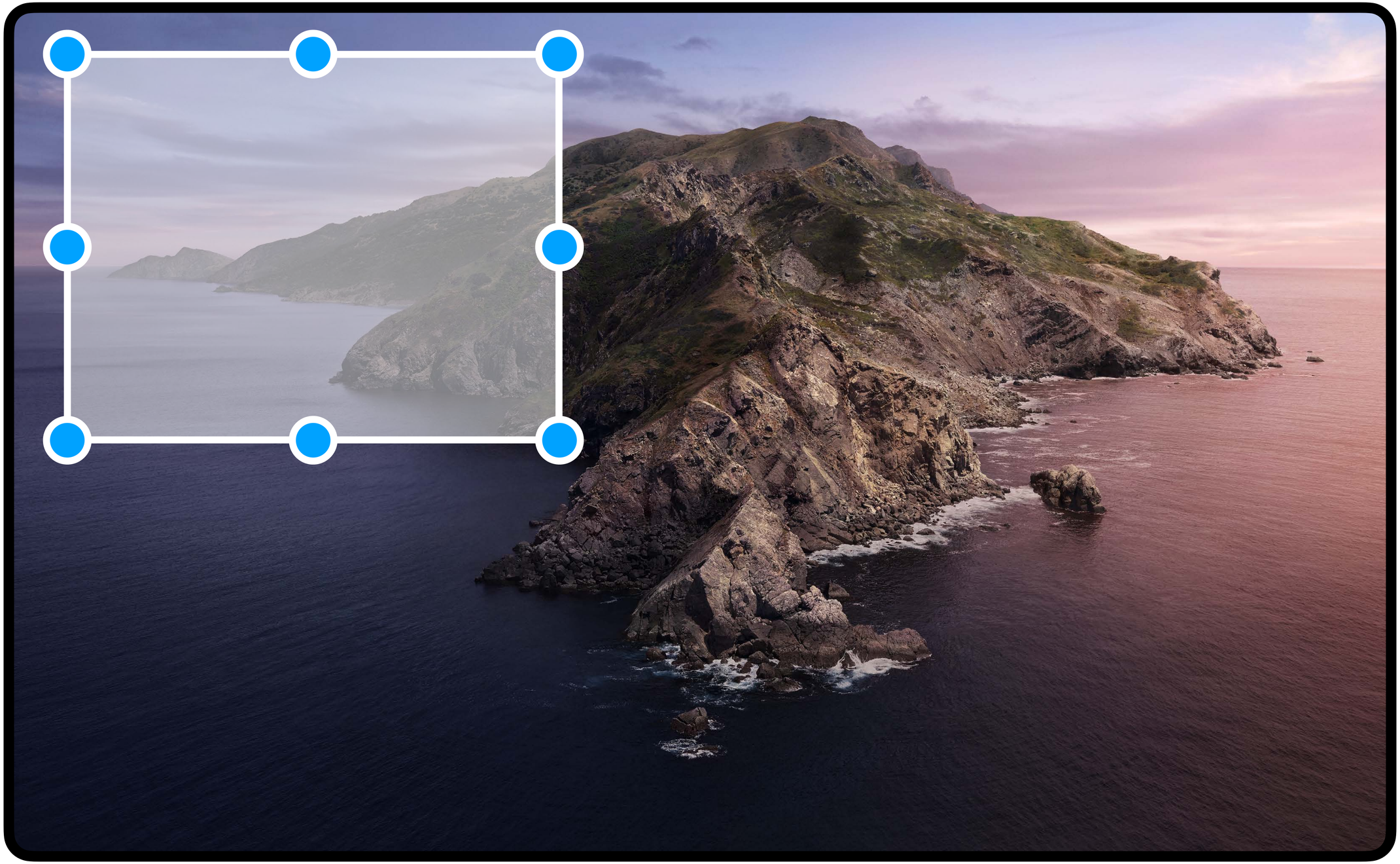
Don’t close the app on MacOS.

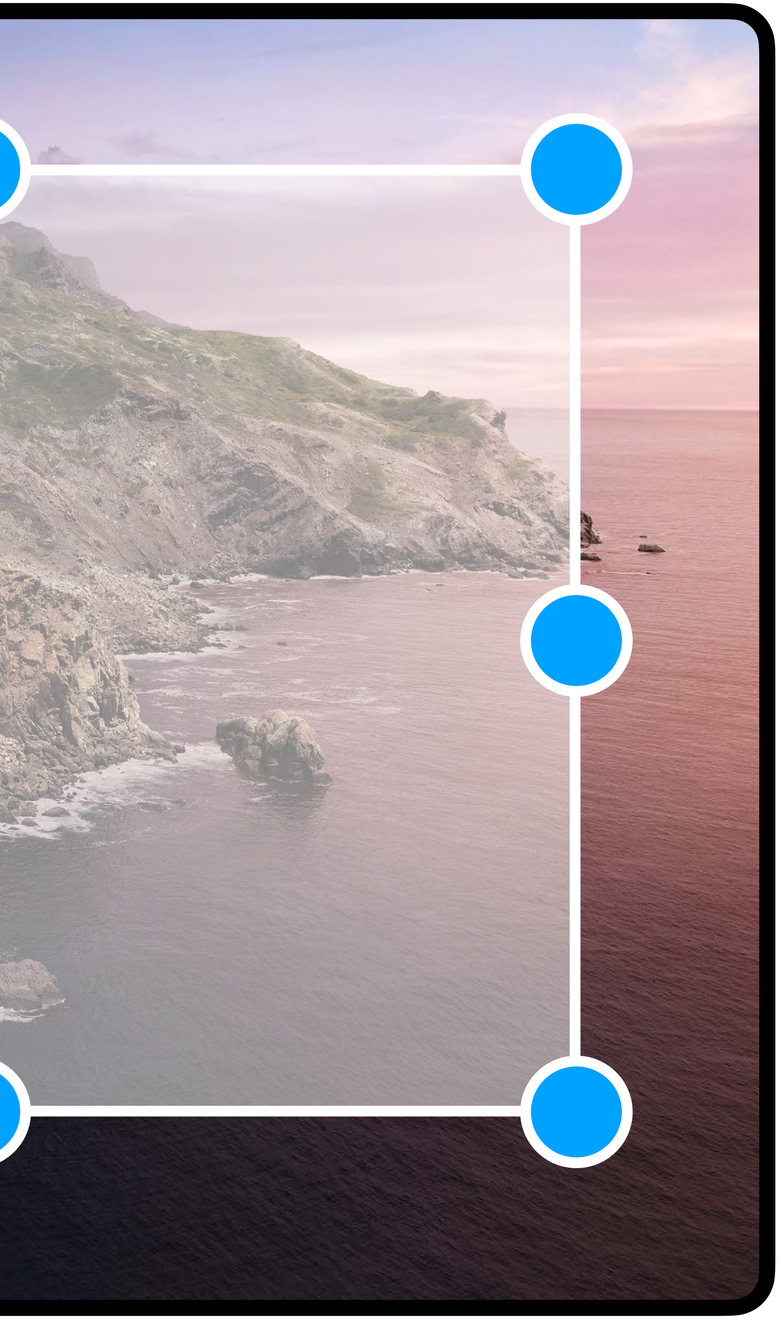
```
const { app, BrowserWindow } = require('electron');

let mainWindow;

function createWindow() {
  mainWindow = new BrowserWindow({
    width: 800,
    height: 600,
    show: false,
  });

  mainWindow.once('ready-to-show', () => { ... })
  mainWindow.loadURL('https://www.conf42.com/js2021');

  mainWindow.on('closed', () => {
    mainWindow = null;
  })
}

app.whenReady().then(createWindow);

app.on('window-all-closed', () => {
  if( process.platform !== 'darwin') {
    app.quit;
  }
})
```

And make sure to recreate the window when the dock icon is clicked.

```
app.on('activate', function () {
  if (mainWindow === null) {
    createWindow()
  }
})
```

3.
Remember user preferences





- **Window dimensions**
- **Window position**
- **Which monitor**
- **If the app is maximised**

- **Window dimensions**
- **Window position**
- **Which monitor**
- **If the app is maximised**

```
let windowState = settings.get('windowState')

['resize', 'move', 'close'].forEach(event ⇒ {
  mainWindow.on(event, () ⇒ {

    windowState.isMaximized = mainWindow.isMaximized()

    if (!windowState.isMaximized) {
      windowState.bounds = mainWindow.getBounds()
    }

    settings.set('windowState', windowState)
  });
});
```

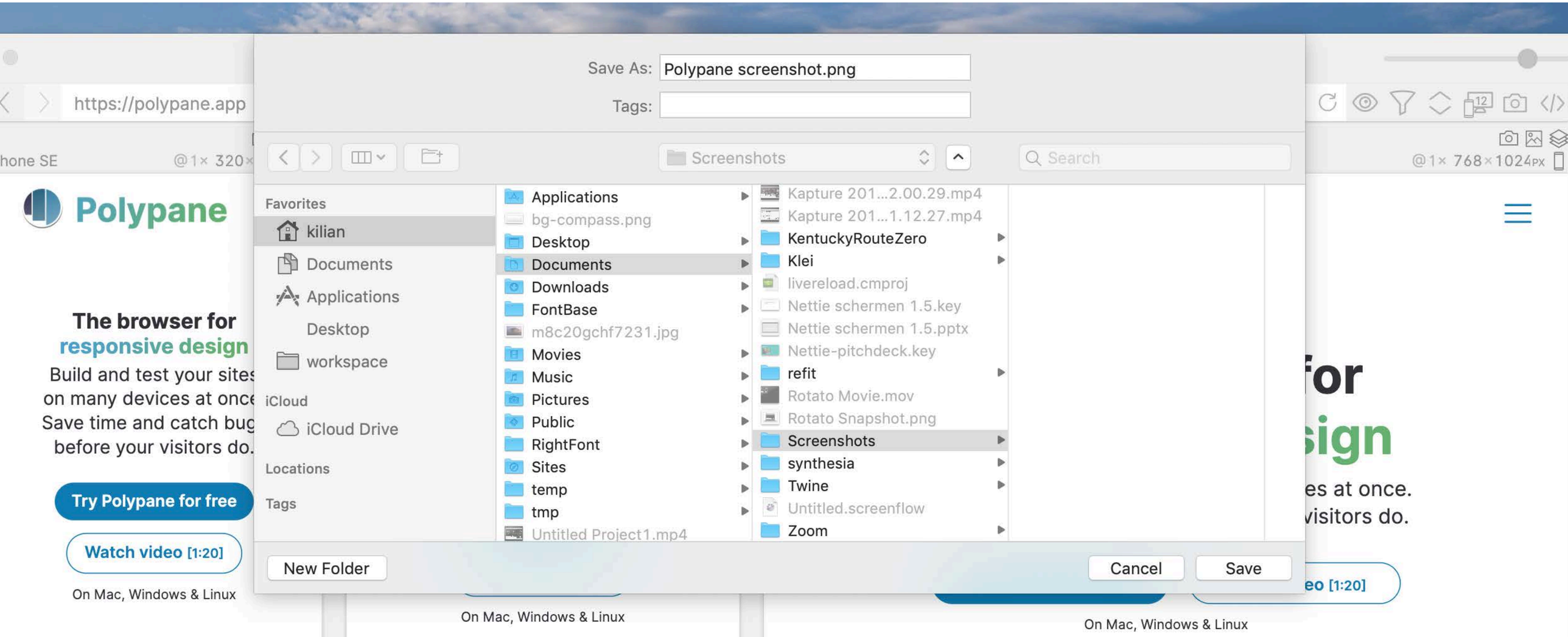
electron-settings

Restore the window position on load

```
let windowState = settings.get('windowState')

mainWindow = new BrowserWindow({
  x: (windowState.bounds && windowState.bounds.x) || undefined,
  y: (windowState.bounds && windowState.bounds.y) || undefined,
  width: (windowState.bounds && windowState.bounds.width) || 800,
  height: (windowState.bounds && windowState.bounds.height) || 600,
  ...
})

mainWindow.once('ready-to-show', () => {
  mainWindow.show()

  if (windowState.isMaximized) {
    mainWindow.maximize()
  }

  mainWindow.focus()
});
```

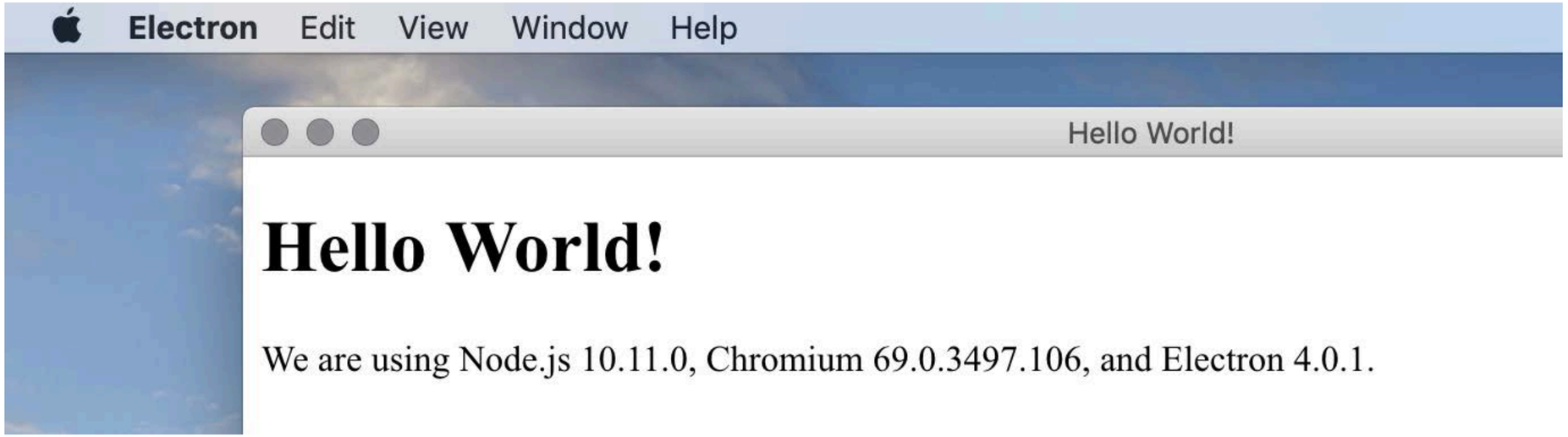
Or use electron-window-state

Keep track of the last used folder



```
dialog.showSaveDialog(
  mainWindow,
  {
    title: 'Save Polypane screenshot',
    defaultPath: `${settings.get('screenshot-path')}/screenshot.png`,
    filters: [{ name: 'Images', extensions: ['png'] }],
  },
  filename => {
    if (filename) {
      setting.set('screenshot-path', path.dirname(filename))
    }
  }
)
```

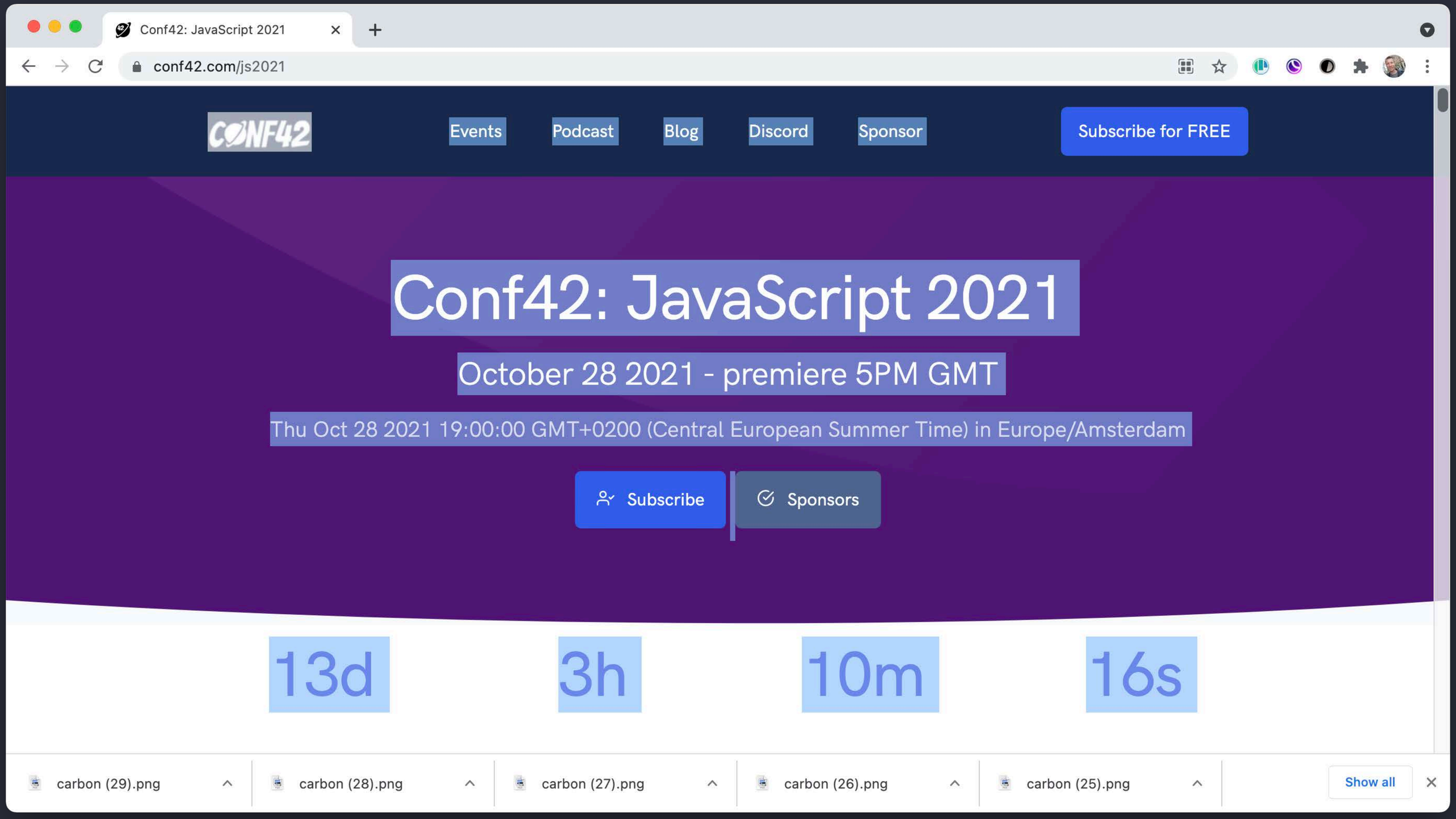
4.
OS-specific menu



npm install electron-create-menu

```
const menuTemplate = [
  {
    label: app.getName(),
    showOn: ["darwin"],
    submenu: [
      { role: "about" },
      SEPARATOR(),
      { role: "services" },
      SEPARATOR(),
      { role: "hide" },
      { role: "hideothers" },
      { role: "unhide" },
      SEPARATOR(),
      { role: "quit" }
    ]
  },
  {
    label: i18nFunc("File"),
    hideOn: ["darwin"],
    submenu: [{ role: "quit" }]
  },
]
```

5.
Text highlighting


Conf42: JavaScript 2021
conf42.com/js2021
CONF42
Events
Podcast
Blog
Discord
Sponsor
Subscribe for FREE
Conf42: JavaScript 2021
October 28 2021 - premiere 5PM GMT
Thu Oct 28 2021 19:00:00 GMT+0200 (Central European Summer Time) in Europe/Amsterdam
Subscribe
Sponsors
13d
3h
10m
16s
carbon (29).png
carbon (28).png
carbon (27).png
carbon (26).png
carbon (25).png
Show all

Untitled.pages — Edited

View | 125% Zoom | Add Page | Insert | Table | Chart | Text | Shape | Media | Comment | Collaborate | Format | Document

Some text in Writer

Text

Default*

Update

Style | Layout | More

Font

SF UI Text

Bold 14 pt

B I U S

Character Styles None

Text Colour

Alignment

Spacing 1,1

Bullets & Lists None

```css
body {
  user-select: none;
}
```

6.
Context menus

```
const { Menu } = require('electron')

mainWindow.webContents.on('context-menu', (e, props) => {
    const menuTemplate = []

    Menu.buildFromTemplate(menuTemplate).popup(mainWindow)
  }
})
```

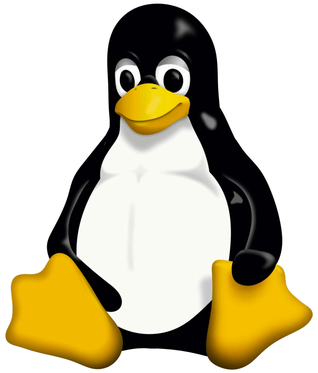
```
const { Menu } = require('electron')

mainWindow.webContents.on('context-menu', (e, props) => {
    const menuTemplate = []

    if (props.isEditable) {
      menuTemplate.push(
        { label: 'Undo',   role: 'undo',  enabled: props.editFlags.canUndo  },
        { label: 'Redo',   role: 'redo',  enabled: props.editFlags.canRedo  },
        { type: 'separator' },
        { label: 'Cut',    role: 'cut',   enabled: props.editFlags.canCut   },
        { label: 'Copy',   role: 'copy',  enabled: props.editFlags.canCopy  },
        { label: 'Paste',  role: 'paste', enabled: props.editFlags.canPaste },
        { type: 'separator' },
        { label: 'Delete',     role: 'delete',    enabled: props.editFlags.canDelete    },
        { label: 'Select all', role: 'selectall', enabled: props.editFlags.canSelectAll }
      )
    }

    Menu.buildFromTemplate(menuTemplate).popup(mainWindow)
  }
})
```

```
const { Menu, shell, clipboard } = require('electron')

mainWindow.webContents.on('context-menu', (e, props) => {
    const menuTemplate = []

    if (props.isEditable) { ... }

    if (props.linkURL) {
      menuTemplate.push(
        { label: 'Open link', click() { shell.openExternal(props.linkURL) } },
        { label: 'Copy link location', click() { clipboard.writeText(props.linkURL) },
      );
    }

    Menu.buildFromTemplate(menuTemplate).popup(mainWindow)
  }
})
```

Or use electron-context-menu

7.
Keyboard shortcuts

Keyboard shortcuts use cmd (⌘)



Keyboard shortcuts use ctrl

```
const { globalShortcut } = require('electron')

globalShortcut.register('backspace', () => { /* ... */ })
```

```
const { globalShortcut } = require('electron')

if (process.platform === 'darwin') {
  globalShortcut.register('cmd+s', () => { /* ... */} )
} else {
  globalShortcut.register('ctrl+s', () => { /* ... */} )
}
```

```
const { globalShortcut } = require('electron')

globalShortcut.register('CmdOrCtrl+s', () => { /* ... */} )
```

CmdOrCtrl

8.
Using system fonts

```
body {

  font-family: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, Helvetica,
Arial, sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol";

}
```

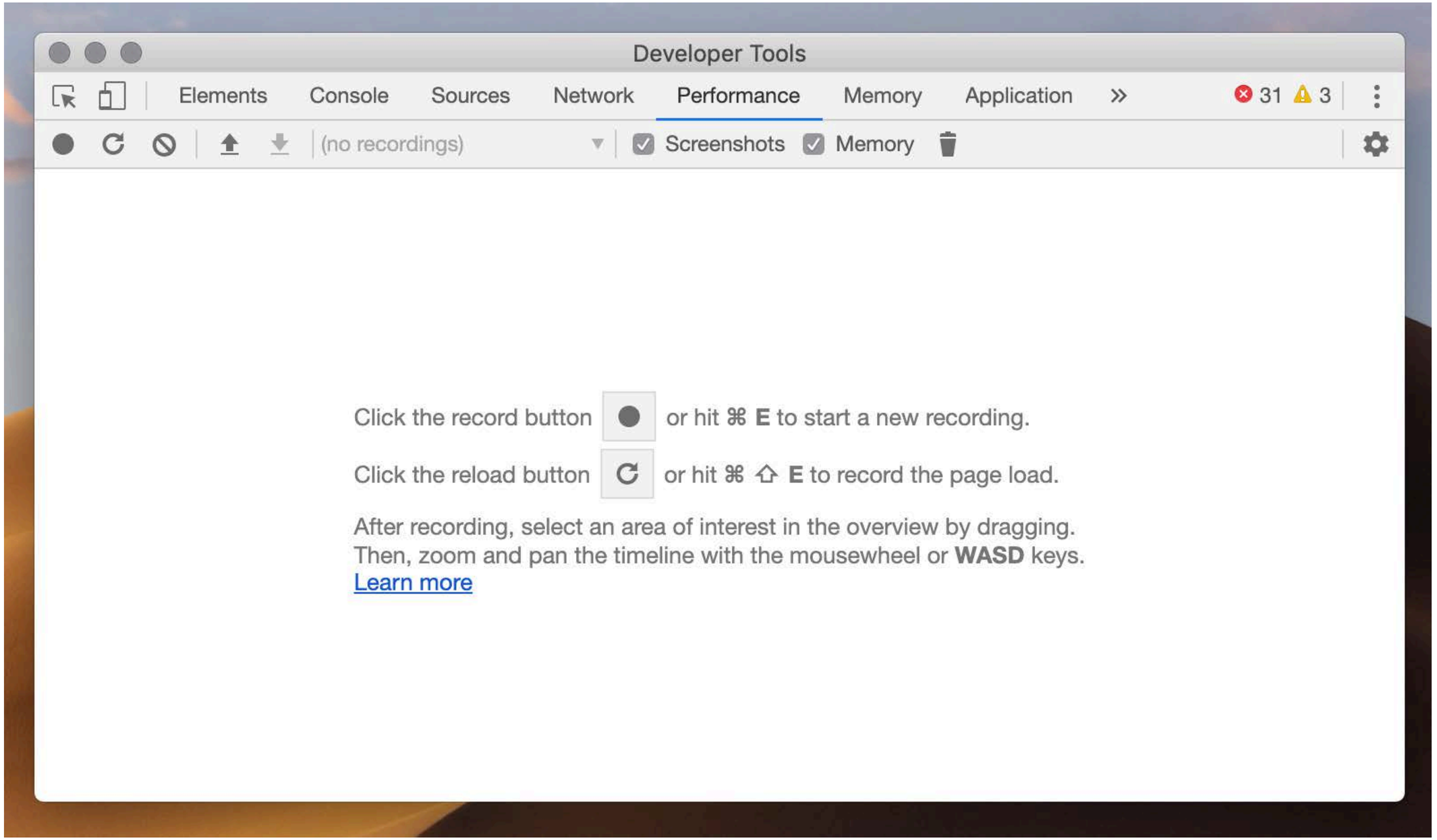
```
body {
  font-family: system-ui;
}
```

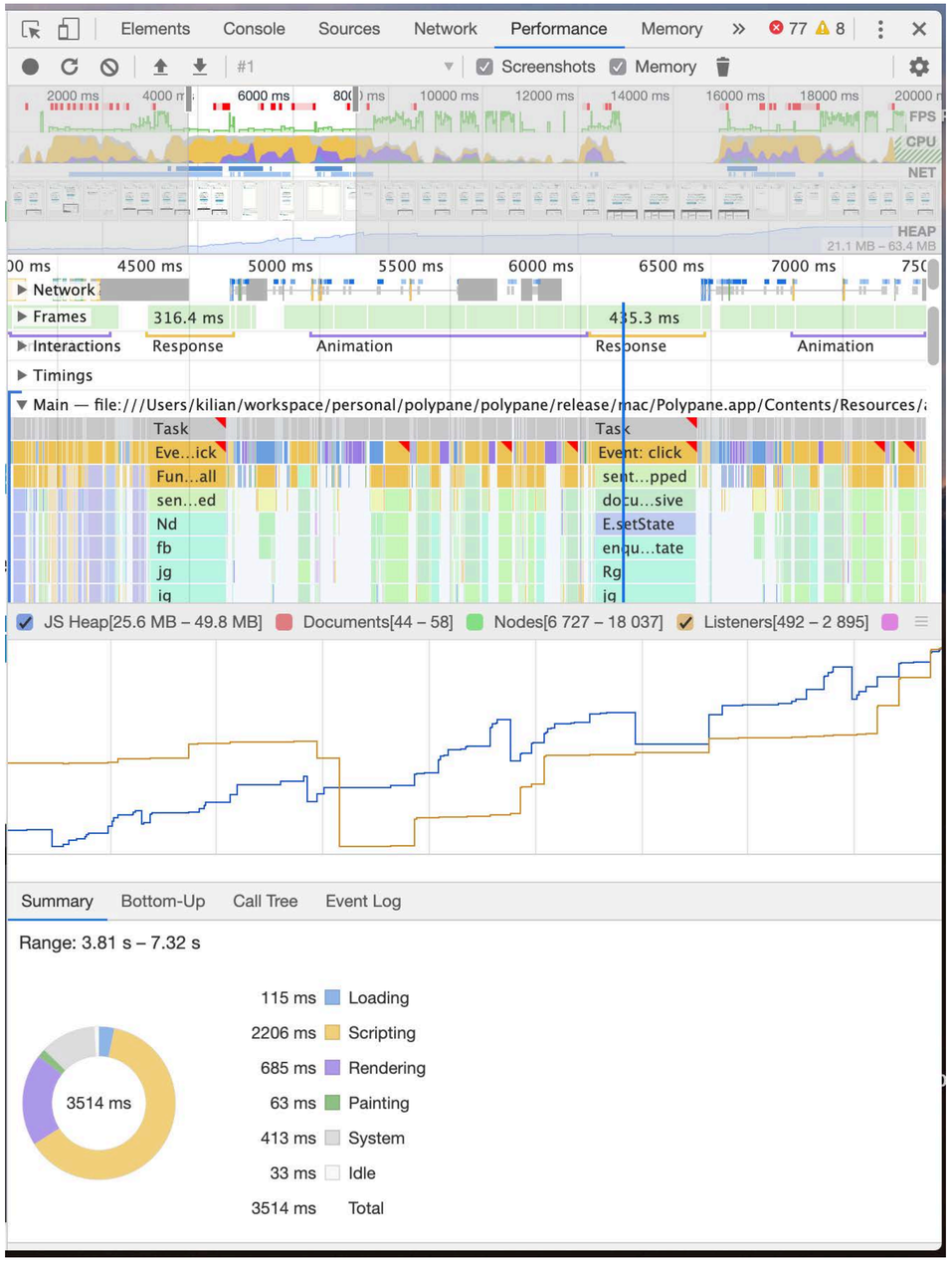
```
@font-face {
  font-family: 'Inter';
  src: url(../font/Inter-Regular.ttf) format('truetype');
  font-weight: 300;
  font-style: normal;
}
```

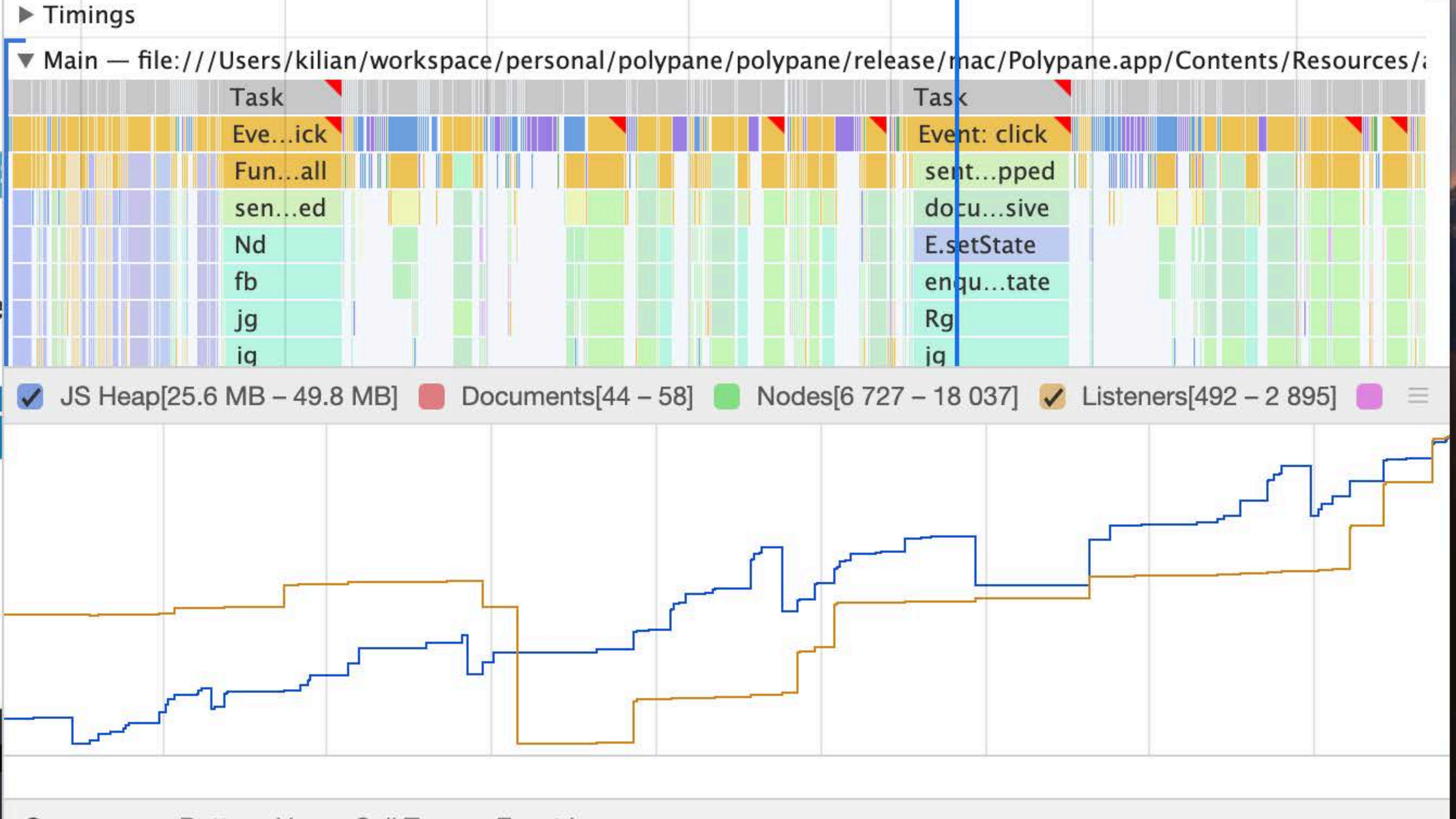
Memory

Memory

https://electronjs.org/docs/tutorial/performance


Developer Tools
Elements
Console
Sources
Network
Performance
Memory
Application
31
3
(no recordings)
Screenshots
Memory
Click the record button or hit ⌘ E to start a new recording.
Click the reload button or hit ⌘ ⇧ E to record the page load.
After recording, select an area of interest in the overview by dragging.
Then, zoom and pan the timeline with the mousewheel or WASD keys.
Learn more


Elements
Console
Sources
Network
Performance
Memory
77
8
#1
Screenshots
Memory
FPS
CPU
NET
HEAP
21.1 MB – 63.4 MB
Network
Frames
316.4 ms
435.3 ms
Interactions
Response
Animation
Response
Animation
Timings
Main — file:///Users/kilian/workspace/personal/polypane/polypane/release/mac/Polypane.app/Contents/Resources/
Task
Task
Eve…ick
Event: click
Fun…all
sen…ed
Nd
fb
jg
ig
JS Heap[25.6 MB – 49.8 MB]
Documents[44 – 58]
Nodes[6 727 – 18 037]
Listeners[492 – 2 895]
Summary
Bottom-Up
Call Tree
Event Log
Range: 3.81 s – 7.32 s
3514 ms
115 ms Loading
2206 ms Scripting
685 ms Rendering
63 ms Painting
413 ms System
33 ms Idle
3514 ms Total


Timings
Main — file:///Users/kilian/workspace/personal/polypane/polypane/release/mac/Polypane.app/Contents/Resources/a
Task
Eve…ick
Fun…all
sen…ed
Nd
fb
jg
ig
Task
Event: click
sent…pped
docu…sive
E.setState
enqu…tate
Rg
ig
JS Heap[25.6 MB – 49.8 MB]
Documents[44 – 58]
Nodes[6 727 – 18 037]
Listeners[492 – 2 895]

```
$ electron --inspect=5858 index.js
```

chrome://inspect

In conclusion

1.
Don’t launch your app like a site

Hide until “ready-to-show”

Use a fitting backgroundColor

2.
Handle window closing like the OS

Keep app running on MacOS, re-open on click.

Closing the (last) window should close the app on Windows and Linux

3.
Remember user preferences

Restore windows to their last position and size

Remember the last used folder

4.
OS specific menu

Use electron-create-menu

5.
Prevent text highlighting

```
body {
  user-select: none;
}
```

6.
Context menus

Right clicking on text should have cut, copy etc.

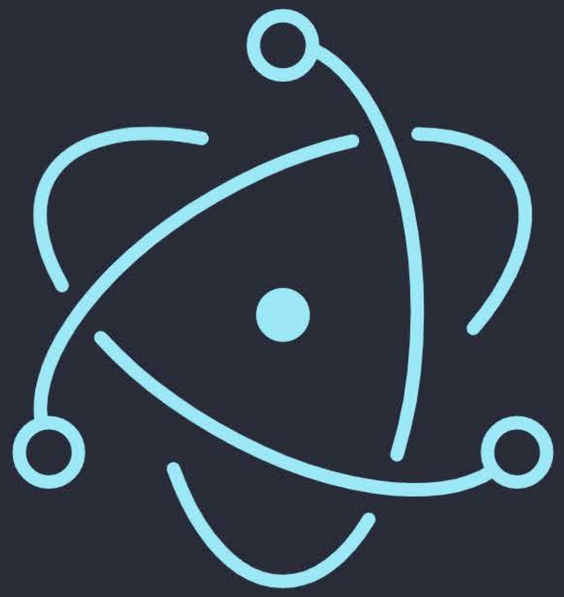
Use contextual info to give extra options for images, links etc.

7.
OS-specific keyboard shortcuts

Use CmdOrCtrl

8.
Use the system font

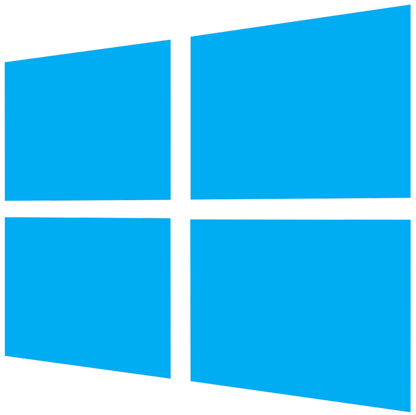
```
body {
  font-family: system-ui;
}
```



electronjs.org

kilianvalkhof.com | @kilianvalkhof | polypane.app

9.
App icons



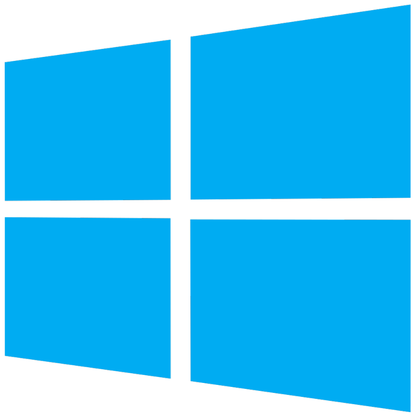
.icns
16x16
32x32
128x128
256x256
512x512
1024x1024

.ico
16x16
24x24
48x48
256x256

.png
1024x1024

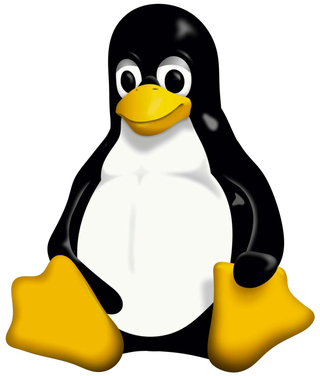
png2icns

16x16
32x32
128x128
256x256
512x512
1024x1024



icotool

16x16
24x24
48x48
256x256



whatever

1024x1024

10.
App layout

Flexbox
CSS Grid

Grid stylesheets:
constraint based css

gss.github.io