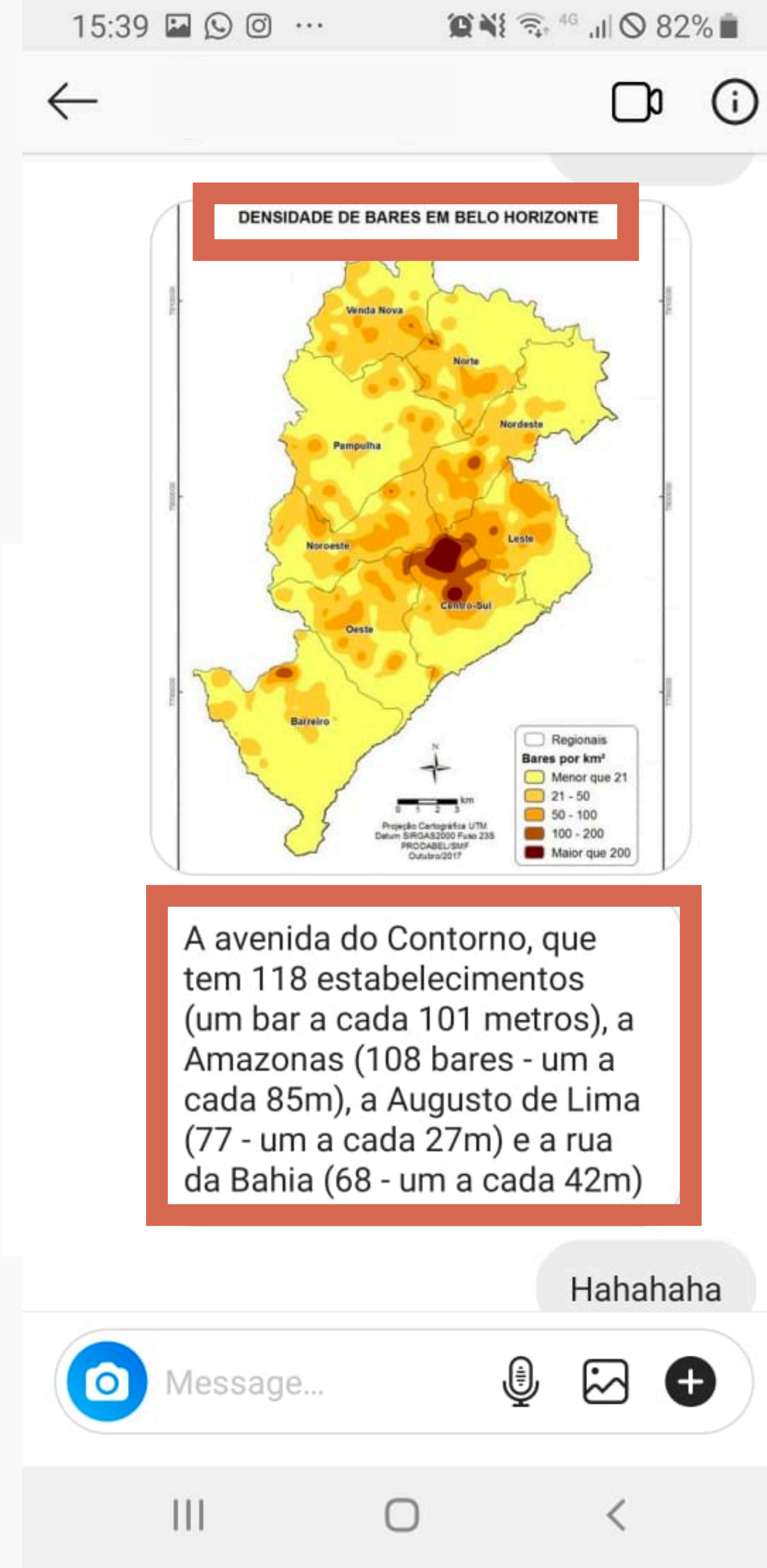
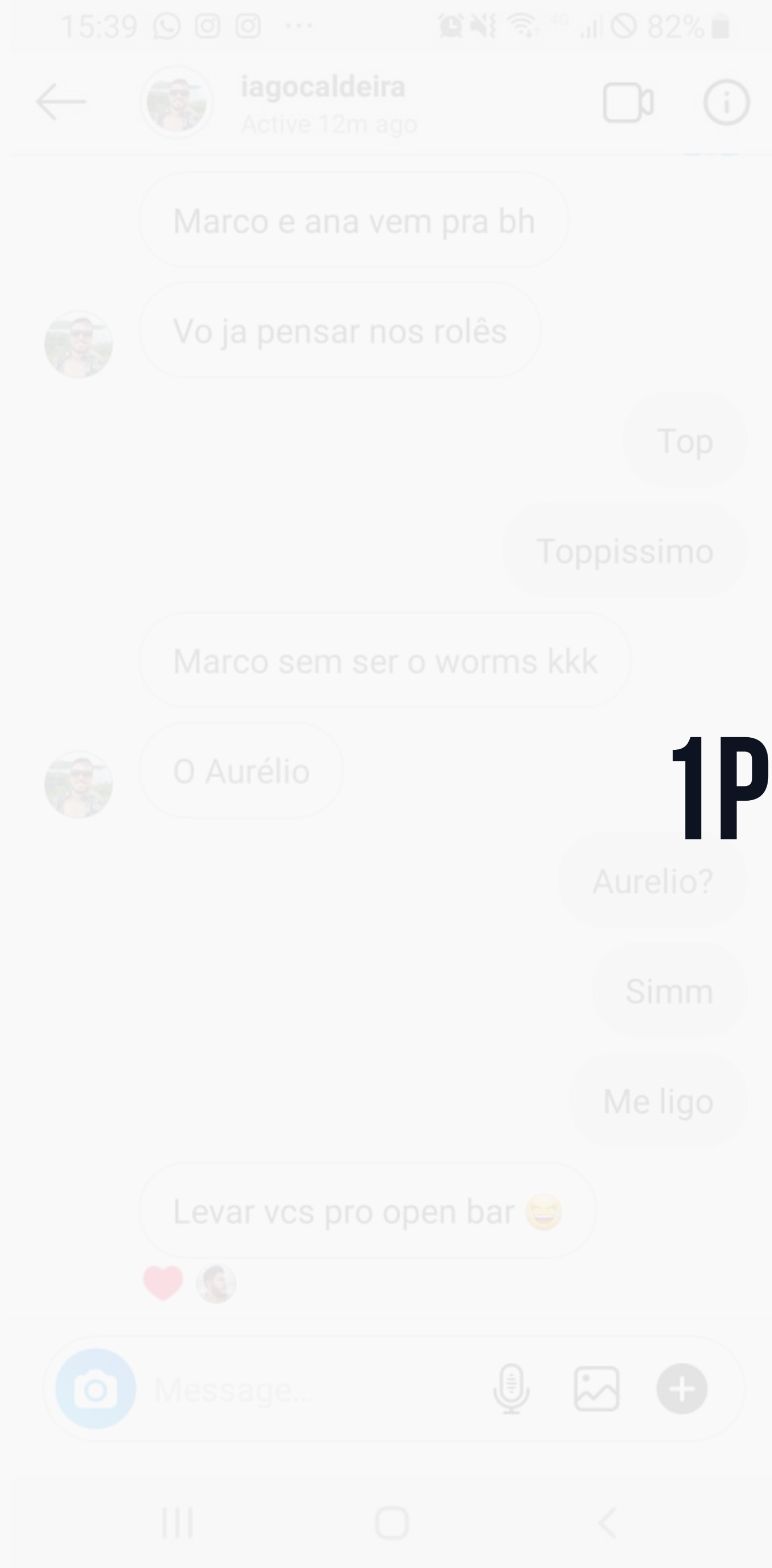




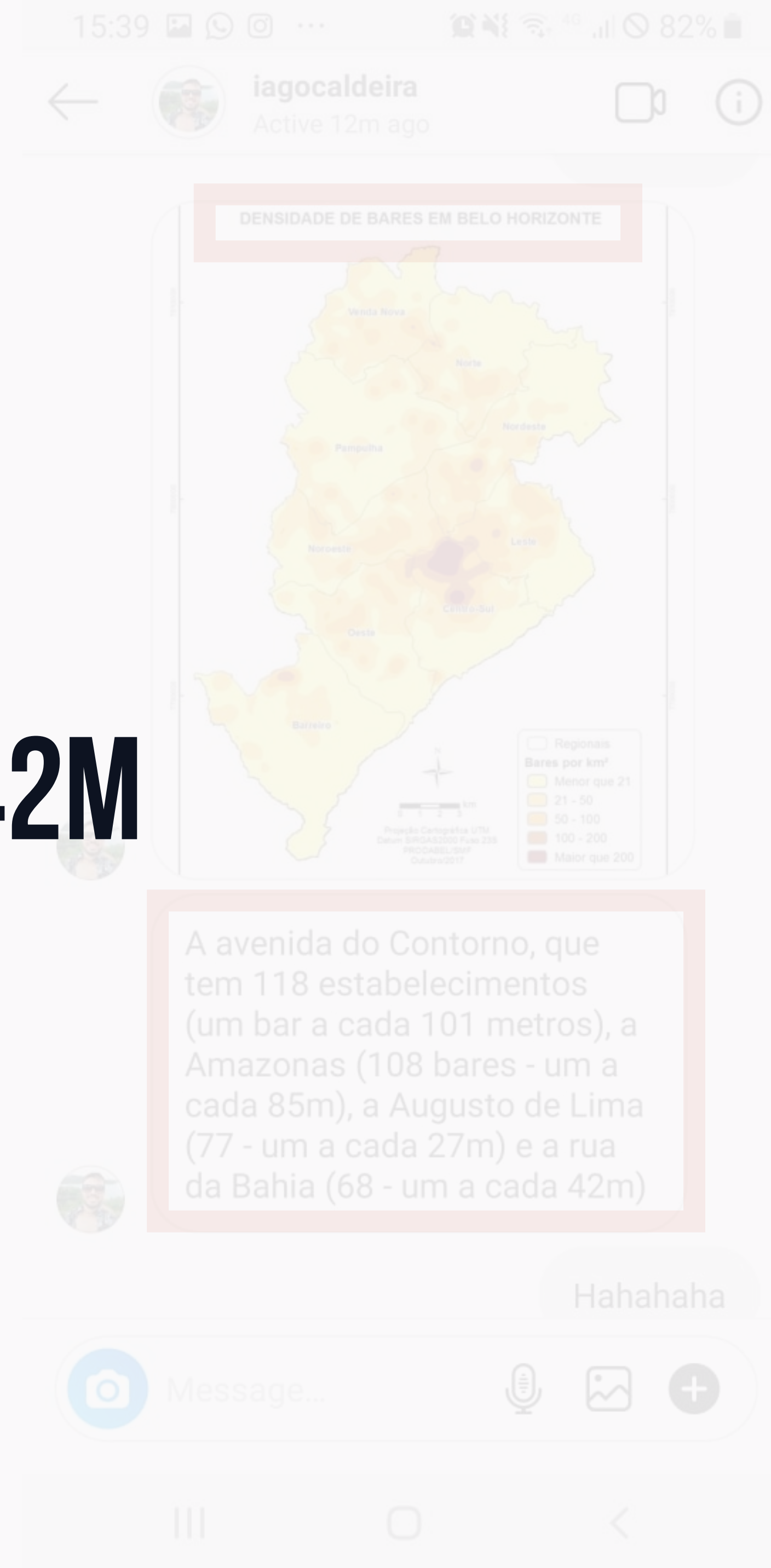
THE STATE OF THE STATE:
REACT STATE MANAGEMENT
IN 2019

@YTHECOMBINATOR





1 PUB/42M



Motorista confunde passarela c x

Matheus

tribunadoceara.uol.com.br/noticias/segurancapublica/motorista-confunde-passarela-c...

UOL HOST PAGSEGURO CURSOS LOJA VIRTUOL

UOL

BUSCA BATE-PAPO EMAIL

tribuna do ceará

NOTÍCIAS EMPREGOS DIVERSÃO FUTEBOLÊS OPINIÃO ESPECIAIS BLOGS PARCEIROS VÍDEOS RÁDIOS E TVS

Motorista confunde passarela com viaduto ao dirigir embriagado

O homem proprietário de um estacionamento da capital ficou detido, mas foi liberado após pagar fiança

Por Lyvia Rocha em **Segurança Pública**
5 de Maio de 2016 às 18:47

Há 2 anos

f G+ t



O motorista que invadiu a passarela do Centro Dragão do Mar de Arte e Cultura, nesta quinta-feira (5), admitiu que dirigia embriagado quando cometeu a infração. O local é ponto turístico de Fortaleza.

Identificado como Jânio Magalhães de Souza, 30 anos, o homem foi detido na 34ª Distrito Policial de Fortaleza, mas foi liberado após pagar fiança.

Proprietário de um estacionamento no Centro da capital, o homem afirmou passar por problemas pessoais e por isso



1º População espanca e amarra ladrão que assaltou ônibus com arma de brinquedo em Fortaleza



2º Taxista é morto após fazer corrida de Fortaleza a Maranguape



3º Homem apontado como fornecedor de drogas na Bolívia é preso em motel no Ceará

Publicidade



Aguardando mais.uol.com.br...

Motorista confunde passarela x

Matheus

tribunadoceara.uol.com.br/noticias/segurancapublica/motorista-confunde-passarela-c...

UOL HOST PAGSEGURO CURSOS LOJA VIRTUOL UOL BUSCA BATE-PAPO EMAIL

tribuna do ceará NOTÍCIAS EMPREGOS DIVERSÃO FUTEBOLÊS OPINIÃO ESPECIAIS BLOGS PARCEIROS VÍDEOS RÁDIOS E TVS

Motorista confunde passarela com viaduto ao dirigir embriagado

O homem proprietário de um estacionamento da capital ficou detido, mas foi liberado após pagar fiança

Por Lyvia Rocha em *Segurança Pública*
5 de Maio de 2016 às 18:47

Há 2 anos

f G+ t



O motorista que invadiu a passarela do Centro Dragão do Mar de Arte e Cultura, nesta quinta-feira (5), admitiu que dirigia embriagado quando cometeu a infração. O local é ponto turístico de Fortaleza.

Identificado como Jânio Magalhães de Souza, 30 anos, o homem foi detido na 34ª Distrito Policial de Fortaleza, mas foi liberado após pagar fiança.

Proprietário de um estacionamento no Centro da capital, o homem afirmou passar por problemas pessoais e por isso

1º População espanca e amarra ladrão que assaltou ônibus com arma de brinquedo em Fortaleza

2º Taxista é morto após fazer corrida de Fortaleza a Maranguape

3º Homem apontado como fornecedor de drogas na Bolívia é preso em motel no Ceará

Publicidade



Aguardando mais.uol.com.br...



Motorista confunde passarela x

tribunadoceara.uol.com.br/noticias/segurancapublica/motorista-confunde-passarela-c...

UOL HOST PAGSEGURO CURSOS LOJA VIRTUOL UOL BUSCA BATE-PAPO EMAIL

tribuna do ceará NOTÍCIAS EMPREGOS DIVERSÃO FUTEBOLÉS OPINIÃO ESPECIAIS BLOGS PARCEIROS VÍDEOS RÁDIOS E TVS f G+ Q

Motorista confunde passarela com viaduto ao dirigir embriagado

O homem proprietário de um estacionamento da capital ficou detido, mas foi liberado após pagar fiança

Por Lyvia Rocha em *Segurança Pública*
5 de Maio de 2016 às 18:47

Há 2 anos

f G+ t



O motorista que invadiu a passarela do Centro Dragão do Mar de Arte e Cultura, nesta quinta-feira (5), admitiu que dirigia embriagado quando cometeu a infração. O local é ponto turístico de Fortaleza.

Identificado como Jânio Magalhães de Souza, 30 anos, o homem foi detido na 34ª Distrito Policial de Fortaleza, mas foi liberado após pagar fiança.

Proprietário de um estacionamento no Centro da capital, o homem afirmou passar por problemas pessoais e por isso

1º População espanca e amarra ladrão que assaltou ônibus com arma de brinquedo em Fortaleza

2º Taxista é morto após fazer corrida de Fortaleza a Maranguape

3º Homem apontado como fornecedor de drogas na Bolívia é preso em motel no Ceará

Publicidade



Aguardando mais.uol.com.br...

Motorista confunde passarela c x

Matheus

tribunadoceara.uol.com.br/noticias/segurancapublica/motorista-confunde-passarela-c...

UOL HOST PAGSEGURO CURSOS LOJA VIRTUOL

UOL

BUSCA BATE-PAPO EMAIL

tribuna do ceará

NOTÍCIAS EMPREGOS DIVERSÃO FUTEBOLÊS OPINIÃO ESPECIAIS BLOGS PARCEIROS VÍDEOS RÁDIOS E TVS

Motorista confunde passarela com viaduto ao dirigir embriagado

O homem proprietário de um estacionamento da capital ficou detido, mas foi liberado após pagar fiança

Por Lyvia Rocha em **Segurança Pública**
5 de Maio de 2016 às 18:47

Há 2 anos

f G+ t



O motorista que invadiu a passarela do Centro Dragão do Mar de Arte e Cultura, nesta quinta-feira (5), admitiu que dirigia embriagado quando cometeu a infração. O local é ponto turístico de Fortaleza.

Identificado como Jânio Magalhães de Souza, 30 anos, o homem foi detido na 34ª Distrito Policial de Fortaleza, mas foi liberado após pagar fiança.

Proprietário de um estacionamento no Centro da capital, o homem afirmou passar por problemas pessoais e por isso



1º População espanca e amarra ladrão que assaltou ônibus com arma de brinquedo em Fortaleza



2º Taxista é morto após fazer corrida de Fortaleza a Maranguape



3º Homem apontado como fornecedor de drogas na Bolívia é preso em motel no Ceará

Publicidade



Aguardando mais.uol.com.br...

Motorista confunde passarela x

Matheus

tribunadoceara.uol.com.br/noticias/segurancapublica/motorista-confunde-passarela-c...

UOL HOST PAGSEGURO CURSOS LOJA VIRTUOL UOL BUSCA BATE-PAPO EMAIL

tribuna do ceará NOTÍCIAS EMPREGOS DIVERSÃO FUTEBOLÊS OPINIÃO ESPECIAIS BLOGS PARCEIROS VÍDEOS RÁDIOS E TVS

Motorista confunde passarela com viaduto ao dirigir embriagado

O homem proprietário de um estacionamento da capital ficou detido, mas foi liberado após pagar fiança

Por Lyvia Rocha em *Segurança Pública*
5 de Maio de 2016 às 18:47

Há 2 anos

f G+ t



O motorista que invadiu a passarela do Centro Dragão do Mar de Arte e Cultura, nesta quinta-feira (5), admitiu que dirigia embriagado quando cometeu a infração. O local é ponto turístico de Fortaleza.

Identificado como Jânio Magalhães de Souza, 30 anos, o homem foi detido na 34ª Distrito Policial de Fortaleza, mas foi liberado após pagar fiança.

Proprietário de um estacionamento no Centro da capital, o homem afirmou passar por problemas pessoais e por isso



1º População espanca e amarra ladrão que assaltou ônibus com arma de brinquedo em Fortaleza



2º Taxista é morto após fazer corrida de Fortaleza a Maranguape



3º Homem apontado como fornecedor de drogas na Bolívia é preso em motel no Ceará

Publicidade



Aguardando mais.uol.com.br...

BRING ME THAT!



APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT UNSTATED

SETSTATEO

REDUX

HOOKS CONTEXT

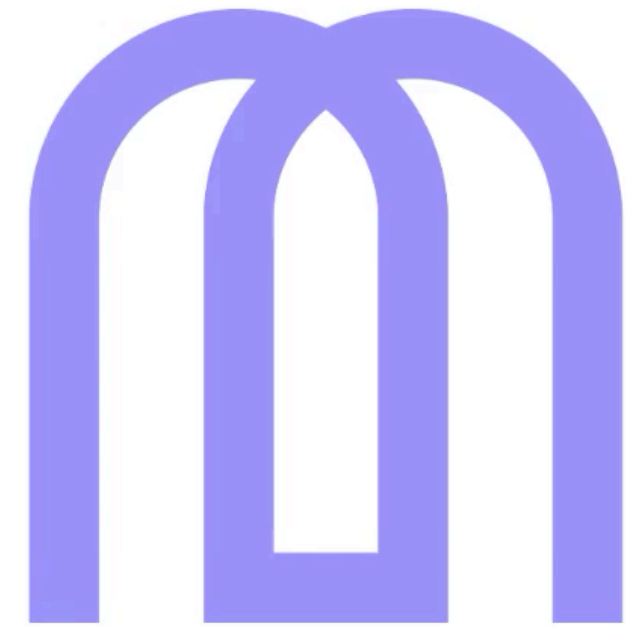
REWORM

RELAY

REACT AUTOMATA FRACTAL

IMMER









ALFA



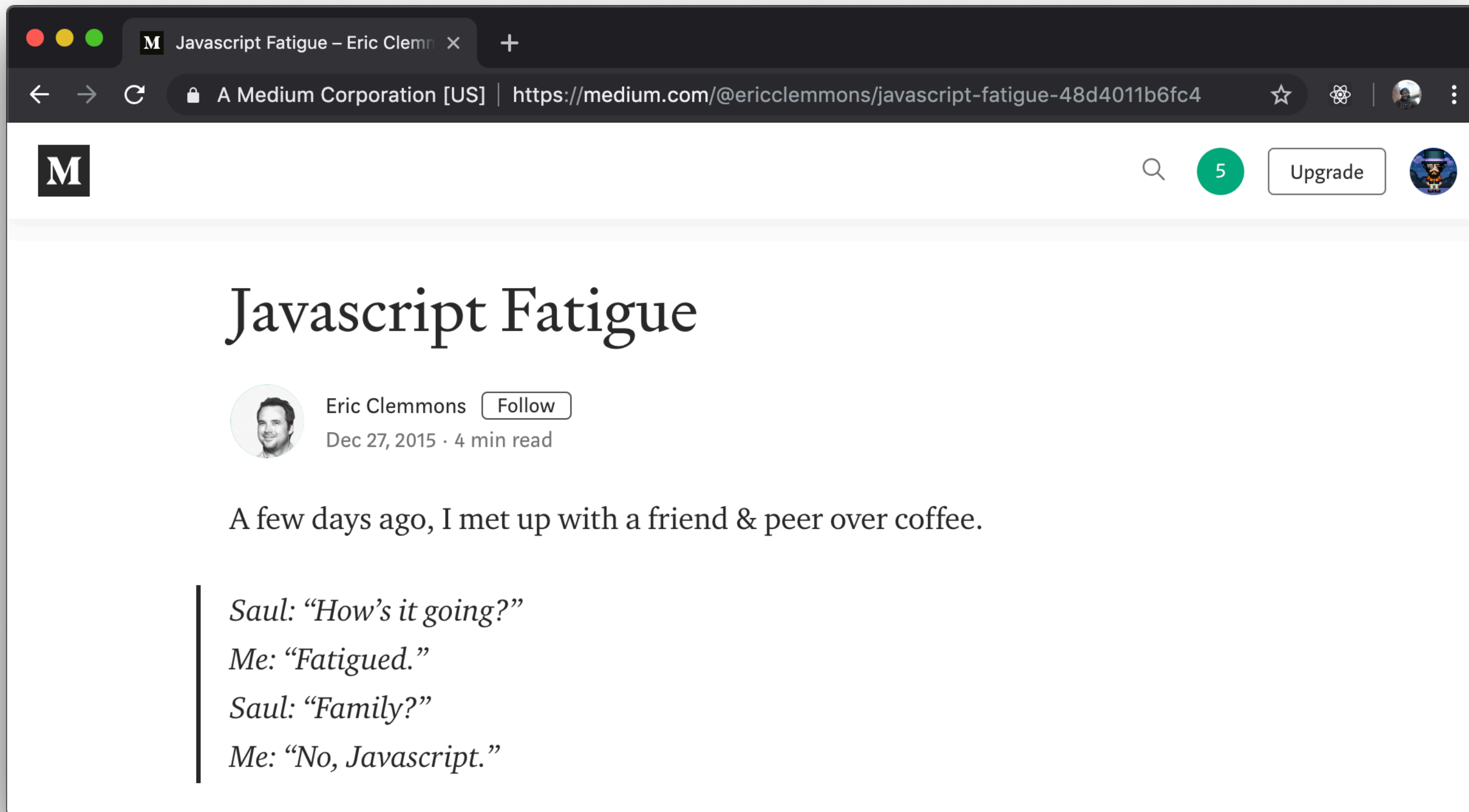
REACT STATE MUSEUM

New recording

One Packing List App Written in:

Technology	React	React Native
setState		
React 16.x Context		
Redux		
MobX		

<https://github.com/GantMan/ReactStateMuseum>



"Javascript Fatigue" by Eric Clemmons

**HAS THE SO-CALLED
JAVASCRIPT FATIGUE
REACHED (REACT)
STATE MANAGEMENT?**

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT UNSTATED

SETSTATEO

REDUX

HOOKS CONTEXT

REWORM

RELAY

REACT AUTOMATA FRACTAL

IMMER

ALFA

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT

UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

FREACTAL

IMMER

ALFA

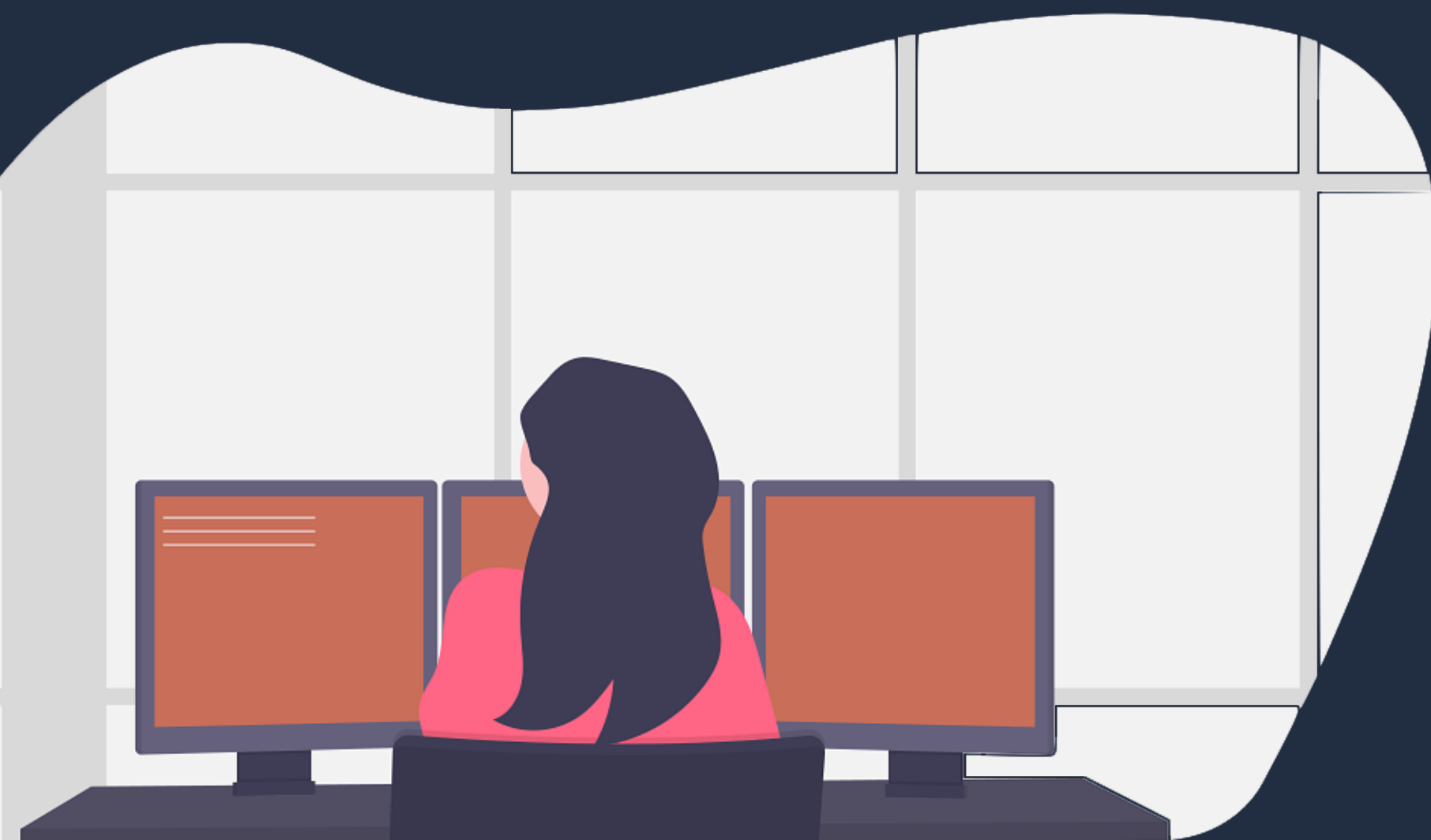
TLDR;

Offload store management logic to a **separate web worker**.

The screenshot shows a web browser window displaying the GitHub repository page for 'developit/stockroom'. The browser's address bar shows the URL 'https://github.com/developit/stockroom'. The repository page includes a search bar, navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore', and a header with the repository name 'developit / stockroom'. Below the header, there are statistics for 'Watch' (26), 'Unstar' (1,521), and 'Fork' (51). A navigation bar shows 'Code' as the active tab, with other options for 'Issues' (7), 'Pull requests' (2), 'Projects' (0), 'Wiki', and 'Insights'. The main content area features a description: 'Offload your store management to a worker easily. https://stockroom.surge.sh', followed by tags for 'unistore', 'worker', 'web-worker', 'flux', 'state-management', and 'state-container'. A summary bar indicates '23 commits', '2 branches', '2 releases', and '4 contributors'. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find File', and 'Clone or download'. The commit history shows a recent merge by 'developit' and two folders: 'demo' (added a year ago) and 'src' (merged 18 days ago).

<https://github.com/developit/stockroom>

DEMO



[@_developit](#) [@mxstbr](#) Very cool, although now I'm curious what kinds of apps you've run into where the store code was pinning the CPU. That hasn't been my experience in my (extremely limited) experience with this kind of thing, but would be cool to hear about your experience.



Sasha Aickin [@xander76](#)

4:02pm - 27 Jan 2018



Jason Miller 🦊🧬 @_developit · 27 de jan de 2018



Em resposta a [@xander76](#) [@mxstbr](#)

None, I use this to avoid memory pressure on the main thread.

🌐 Traduzir Tweet



Alexey Migutsky @mr_mig_by · 28 de jan de 2018



Em resposta a [@xander76](#) [@_developit](#) [@mxstbr](#)

We have that kind of code in Microsoft To-Do. It's a fat client, some features are CPU-bound for high number of to-dos (think full-text search as an example)

🌐 Traduzir Tweet



Federico Zivolo @FezVrasta · 27 de jan de 2018



Em resposta a [@xander76](#) [@_developit](#) [@mxstbr](#)

where I work we use them to compute stuff needed to render canvas/webgl, I think it's the most common use case

🌐 Traduzir Tweet





Jason Miller 🐱🌿 @_developit · 27 de jan de 2018



Em resposta a @xander76 @mxstbr

None, I use this to avoid memory pressure on the main thread.

🌐 Traduzir Tweet



3



Alexey Migutsky @mr_mig_by · 28 de jan de 2018



Em resposta a @xander76 @_developit @mxstbr

We have that kind of code in Microsoft To-Do. It's a fat client, some features are CPU-bound for high number of to-dos (think full-text search as an example)

🌐 Traduzir Tweet



4



Federico Zivolo @FezVrasta · 27 de jan de 2018



Em resposta a @xander76 @_developit @mxstbr

where I work we use them to compute stuff needed to render canvas/webgl, I think it's the most common use case

🌐 Traduzir Tweet



3



HIGHLIGHTS:

→ + Performance

→ Good for **CPU-bound** store mutations

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT

UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

FREACTAL

IMMER

ALFA

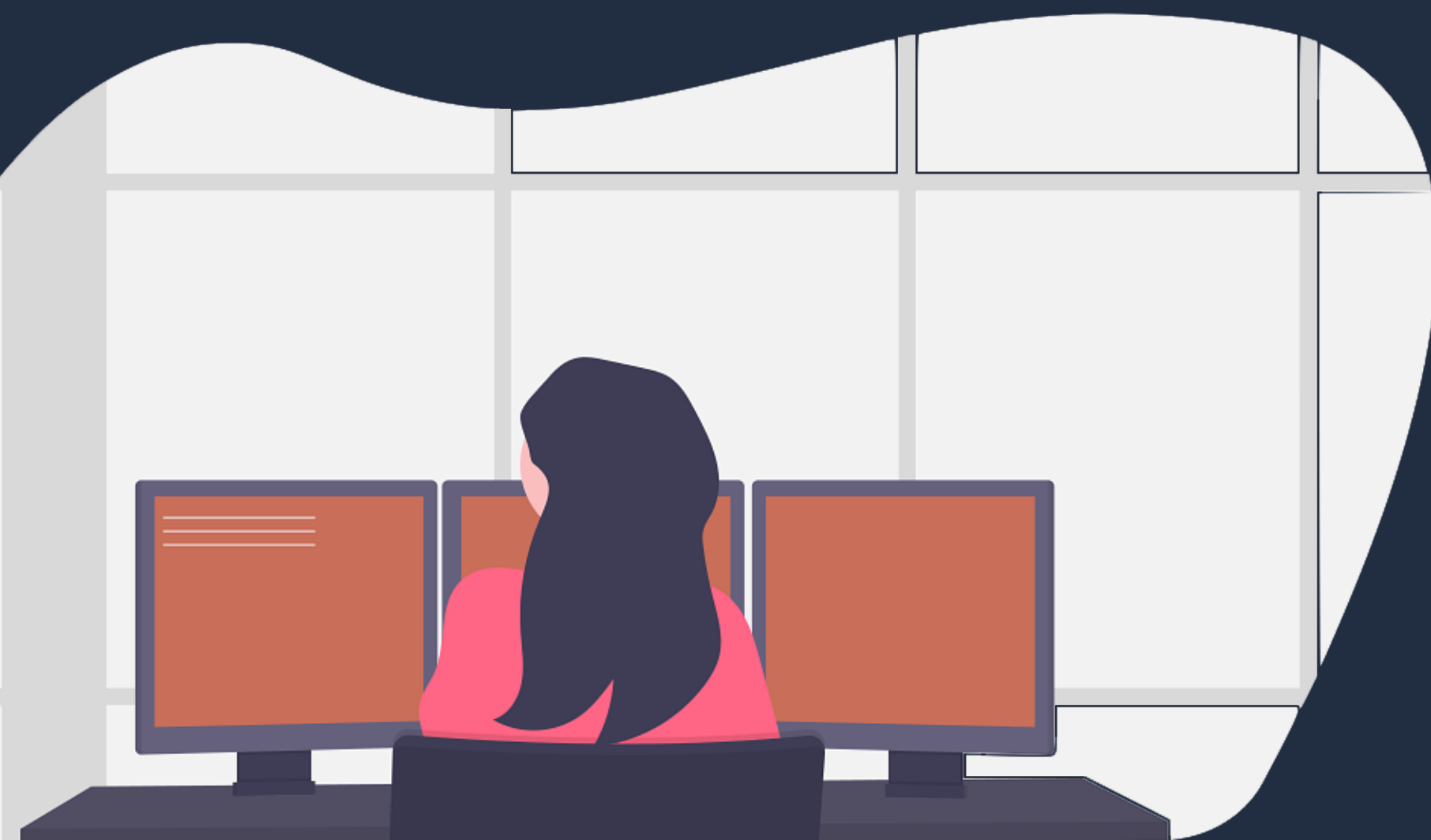
TLDR;

Declaratively describing the behaviour of your application

FEATURING:

- A **list of states**
- **One** initial state
- A list of events that trigger **transitions**

DEMO



HIGHLIGHTS:

- **Preciseness** of specs
- **Correctness** of code
- **Less** unimportant state / **Less** if clauses
- Leverage **knowledge from non-coders** too

CONS:

- Surely an **overkill** for super simple components
- **Lack of familiarity** from developers and designers
- **Small** ecosystem

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT

UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

FREACTAL

IMMER

ALFA

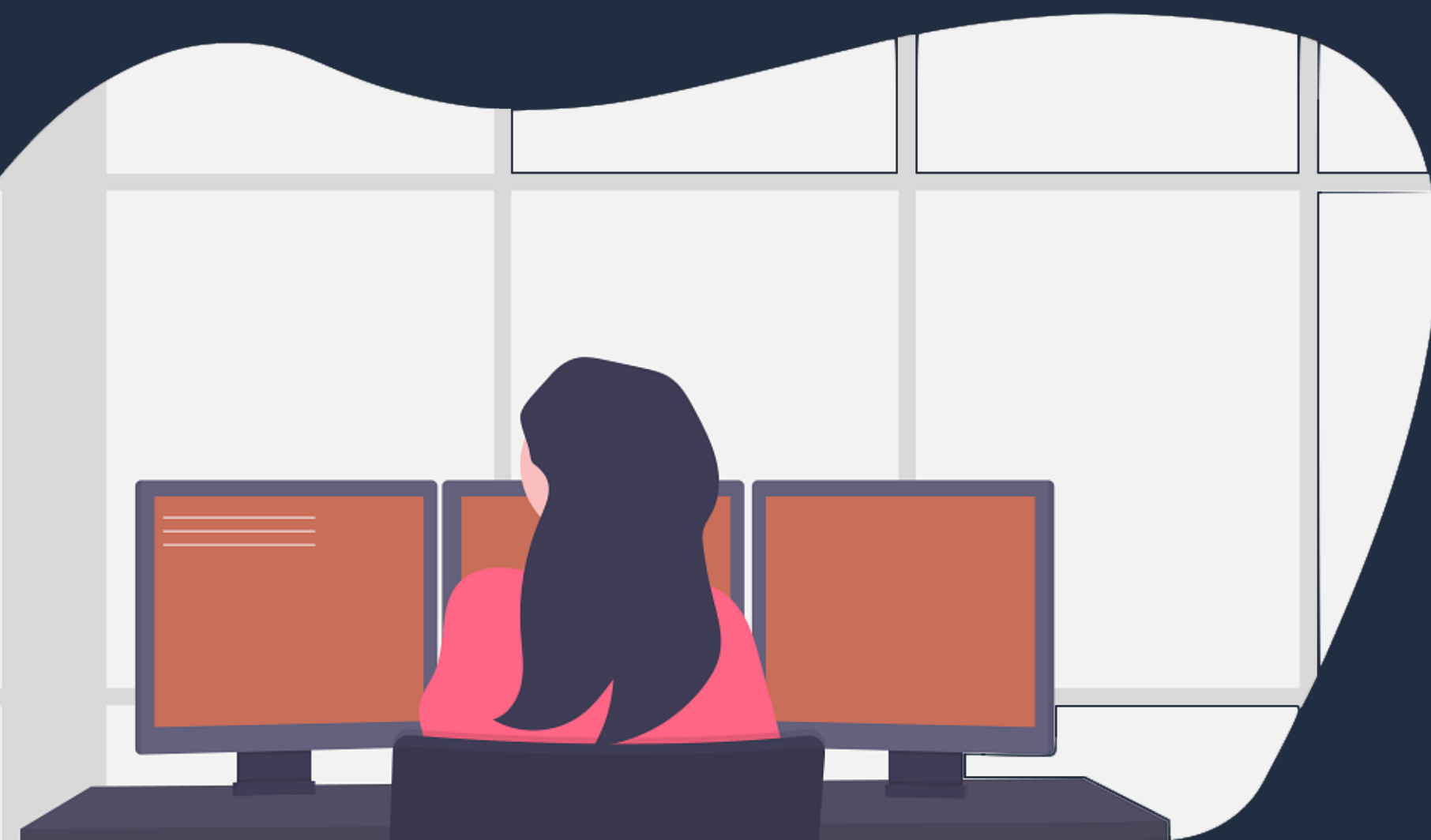
TLDR;

Combine your **local** and **remote** state management

KEY CONCEPTS:

- **DEFAULTS:** Your base state. What you start with.
- **RESOLVERS:** Where all the magic happens to retrieve and update your local data in the Apollo cache.

DEMO



HIGHLIGHTS:

- Getting Apollo set up just for state management can be a **bit of work**
- A **natural way** of querying and mutating state

CONS:

- Surely an **overkill** for apps that don't heavily depend on remote state

A "must-have" for SQL lovers



Query languages for client side state

Making simple look hard

O RLY?

Mr. JavaBean

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT

UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

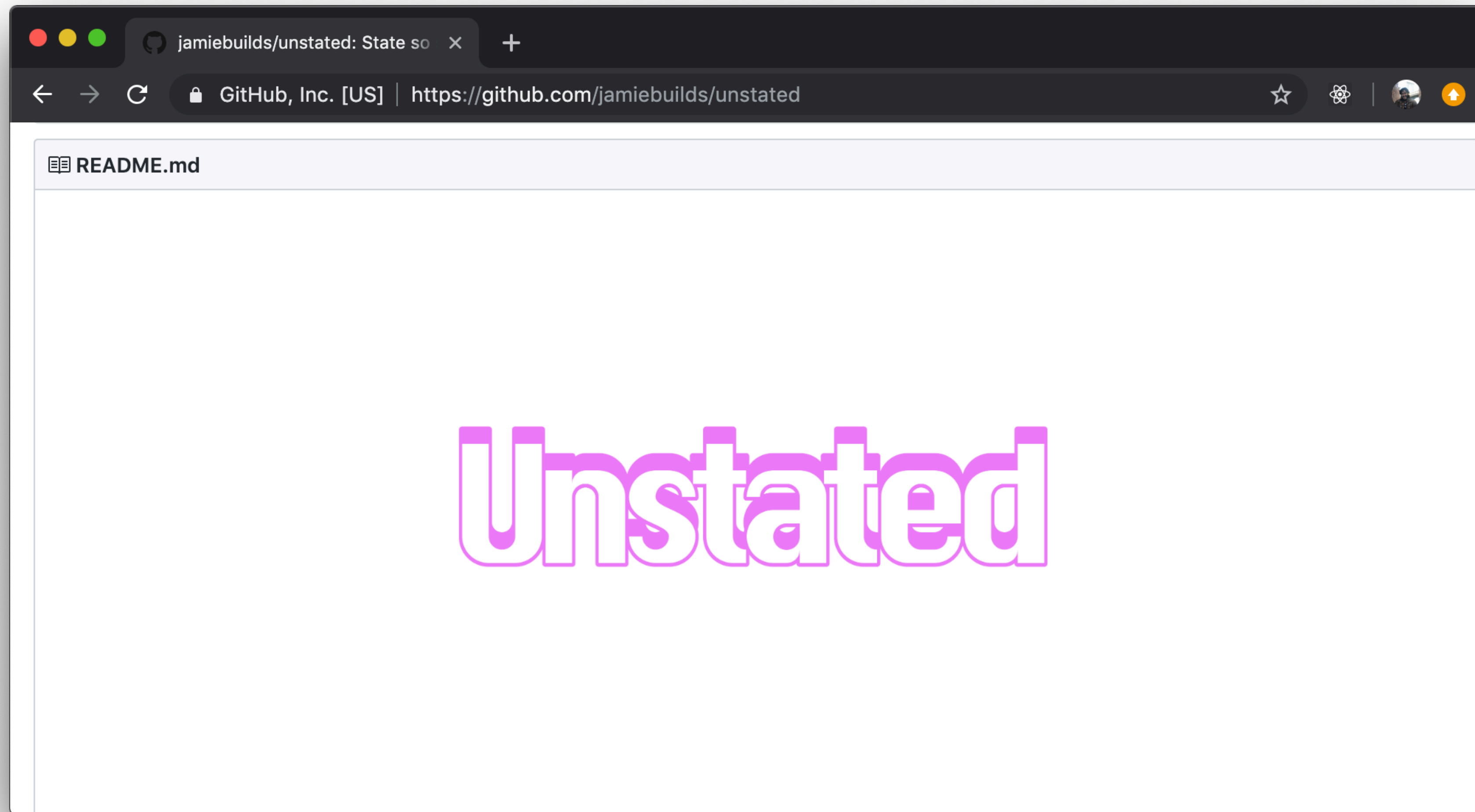
FREACTAL

IMMER

ALFA

TLDR;

Redux best practices **without the boilerplate.**



<https://github.com/jamiebuilds/unstated>

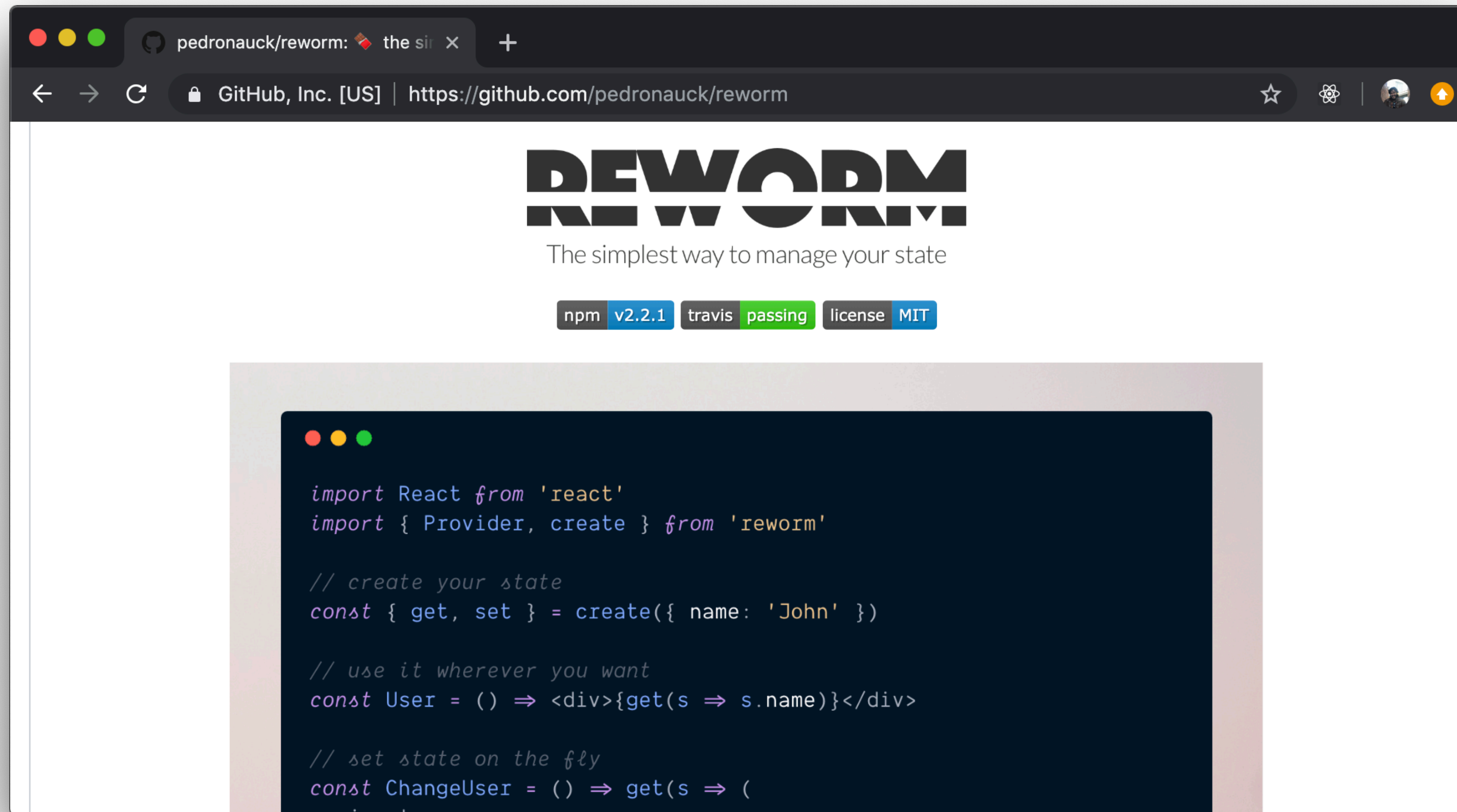
Unstated is a breath of fresh air for React state management, and the first time I've actually enjoyed using one.

[github.com/jamiebuilds/un...](https://github.com/jamiebuilds/unstated)



Sindre Sorhus @sindresorhus

4:03pm - 1 Mar 2018



<https://github.com/pedronauck/reworm>



```
import React from 'react'
import { Provider, create } from 'redux'

const { set, get } = create({ name: 'John' })

class App extends React.Component {
  componentDidMount() {
    set(prev => ({ name: 'Peter' + prev.name }))
  }
  render() {
    return (
      <Provider>
        <div>{get(s => s.name)}</div>
      </Provider>
    )
  }
}
```

HIGHLIGHTS:

→ - Boilerplate

→ + Simplicity

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT

UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

FREACTAL

IMMER

ALFA

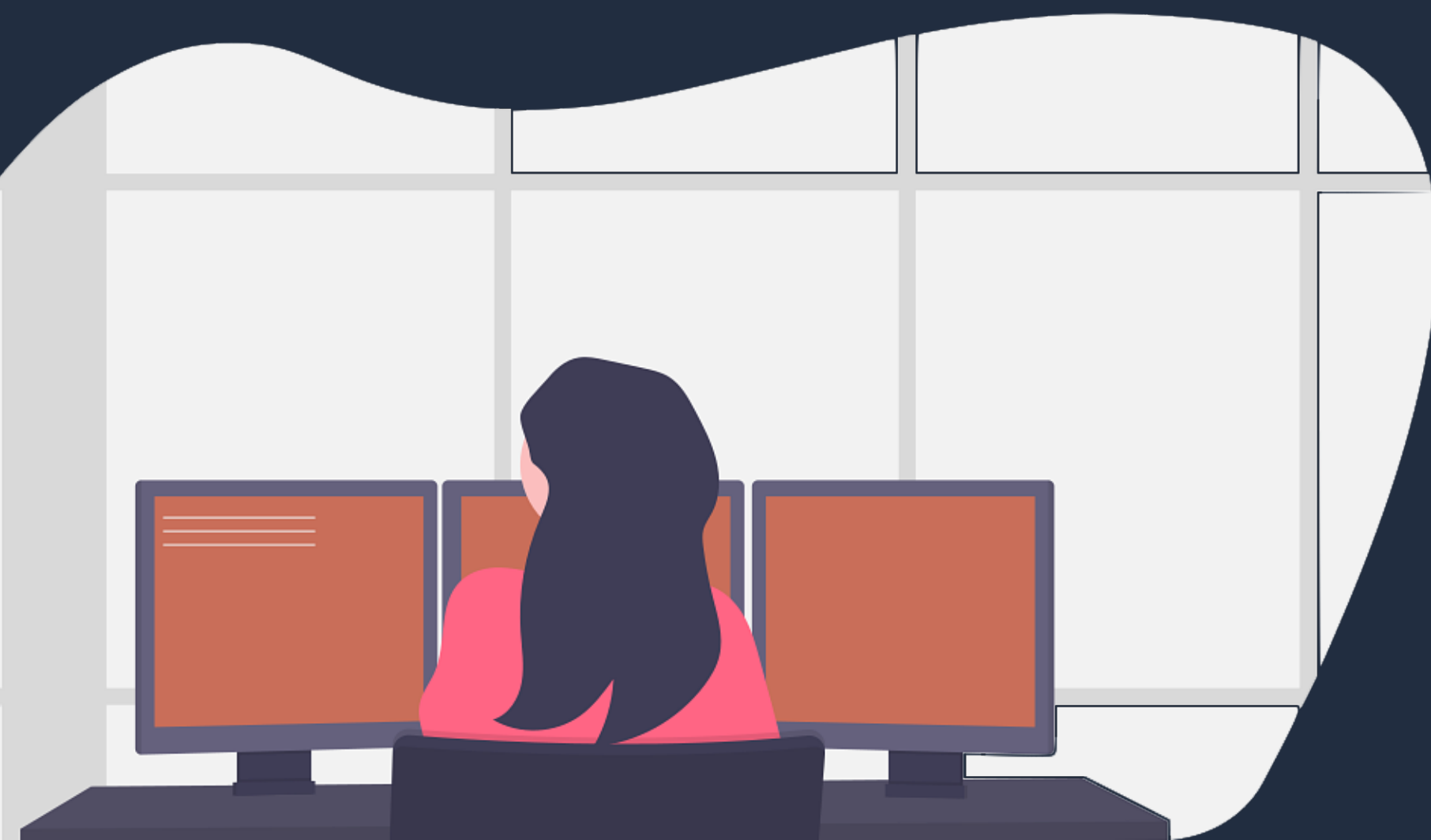
TLDR;

Recreate Redux in our React app **without actually installing redux and react-redux**

KEY CONCEPTS:

- **CONTEXT:** Provides a way to pass data through the component tree without having to pass props down manually at every level.
- **HOOKS:** Let you use state and other React features without writing a class

DEMO



HIGHLIGHTS:

- **Less boilerplate** than redux itself
- **No third-party dependencies**

CONS:

- Still carries some of **redux boilerplate**
- You might face a few **perf issues**

Split contexts that don't change together

```
function Button() {  
  let theme = useContext(ThemeContext);  
  // The rest of your rendering logic  
  return <ExpensiveTree className={theme} />;  
}
```

Split your component in two, put **memo** in between

```
function Button() {  
  let appContextValue = useContext(AppContext);  
  let theme = appContextValue.theme; // Your "selector"  
  return <ThemedButton theme={theme} />  
}
```

```
const ThemedButton = memo(({ theme }) => {  
  // The rest of your rendering logic  
  return <ExpensiveTree className={theme} />;  
});
```

One component with **useMemo** inside

```
function Button() {  
  let appContextValue = useContext(AppContext);  
  let theme = appContextValue.theme; // Your "selector"  
  
  return useMemo(() => {  
    // The rest of your rendering logic  
    return <ExpensiveTree className={theme} />;  
  }, [theme])  
}
```

**HAS THE SO-CALLED
JAVASCRIPT FATIGUE
REACHED (REACT)
STATE MANAGEMENT?**

HAS THE SO-CALLED
JAVASCRIPT FATIGUE
REACHED (REACT)
STATE MANAGEMENT?

MAYBE! 🙄

**BUT SO HAS THE
VANILLA WAY OF
DOING THINGS!**

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT

UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

FREACTAL

IMMER

ALFA

The screenshot shows the GitHub repository page for `pauldijou/redux-act`. The repository is described as "An opinionated lib to create actions and reducers for Redux". It has 1,420 users using it, 15 watchers, 1,353 stars, and 65 forks. The repository has 129 commits, 1 branch, 27 releases, and 17 contributors. The license is Apache-2.0. The current branch is `master`. The repository contains several files and folders, including `examples`, `src`, `test`, `.babelrc`, and `.gitignore`.

File/Folder	Description	Time
<code>examples</code>	[AT] - Updated to React 16 and prop-types, renamed JSX containing fil...	last year
<code>src</code>	Added string action type test (#68)	last year
<code>test</code>	Fix meta reducer for complex actions with params	8 days ago
<code>.babelrc</code>	Update dependencies	17 days ago
<code>.gitignore</code>	Update dependencies	17 days ago

<https://github.com/pauldijou/redux-act>


```
const serializeTodo = createAction('SERIALIZE_TODO');  
serializeTodo(1);  
// return { type: 'SERIALIZE_TODO', payload: 1 }
```

```
const loading = createReducer({}, initialState.loading)
  .on(actions.fetch, () => true)
  .on(actions.setQuery, () => true)
```

```
const ids = createReducer({}, initialState.ids)
  .on(actions.setIds, (state, payload) => [...payload])
  .on(actions.delete, (state, payload) => {
    return [...state.filter((id) => id !== payload)];
  });
```

HIGHLIGHTS:

- **Less boilerplate** than plain redux
- **Piping** syntax for reducers

APOLLO LINK STATE

MOBX

STOCKROOM

REDUX-ACT UNSTATED

SETSTATEO

REDUX

HOOKS

CONTEXT

REWORM

RELAY

REACT AUTOMATA

FREACTAL

IMMER

ALFA

 **INTERMISSION** 

IMMUTABILITY

WHY:

- - Breakable
- + Debuggable
- + Performant

WHY:

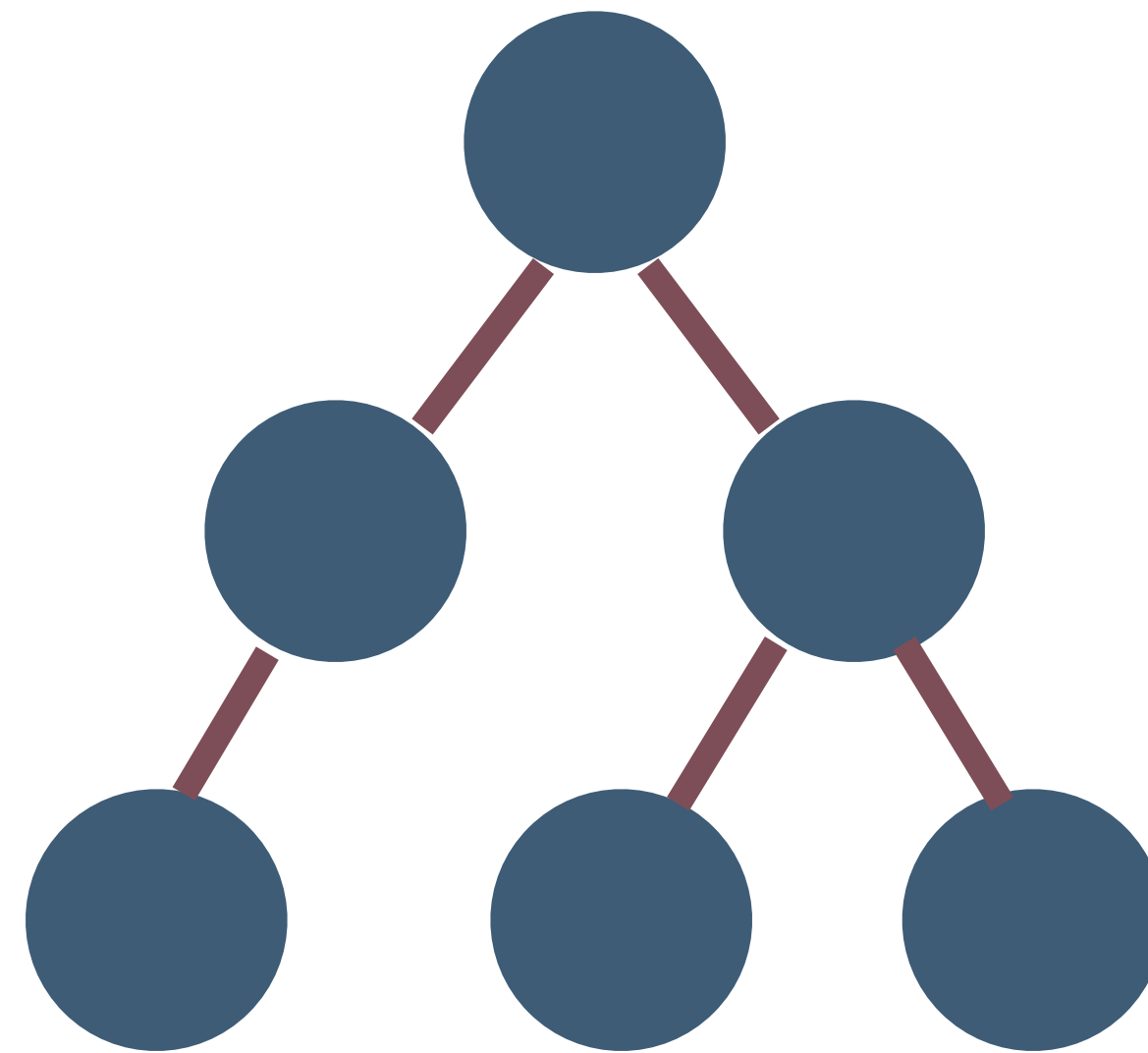
→ - Breakable

→ + Debuggable

→ + Performant

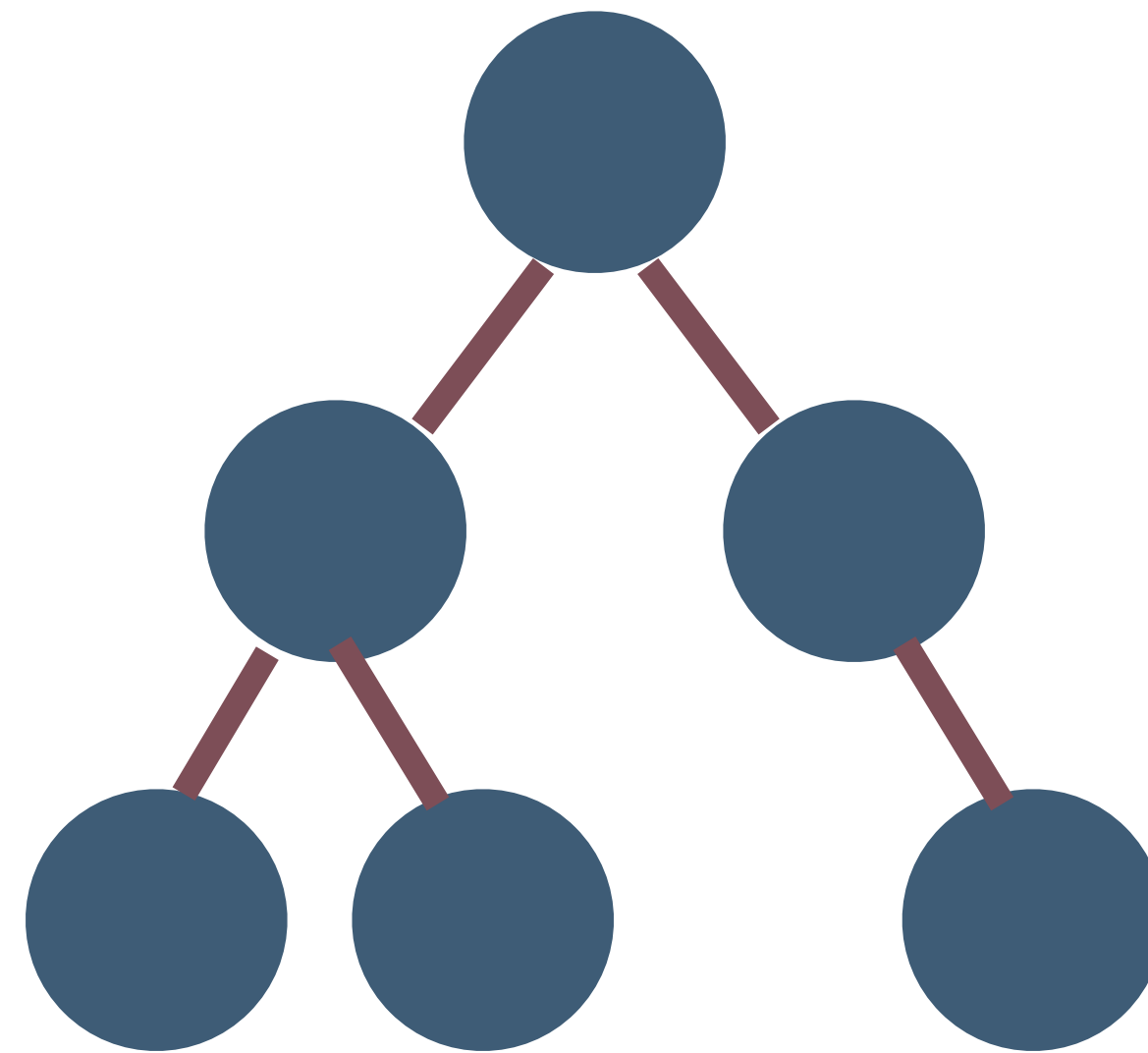
WHY:

- - Breakable
- + Debuggable
- + Performant



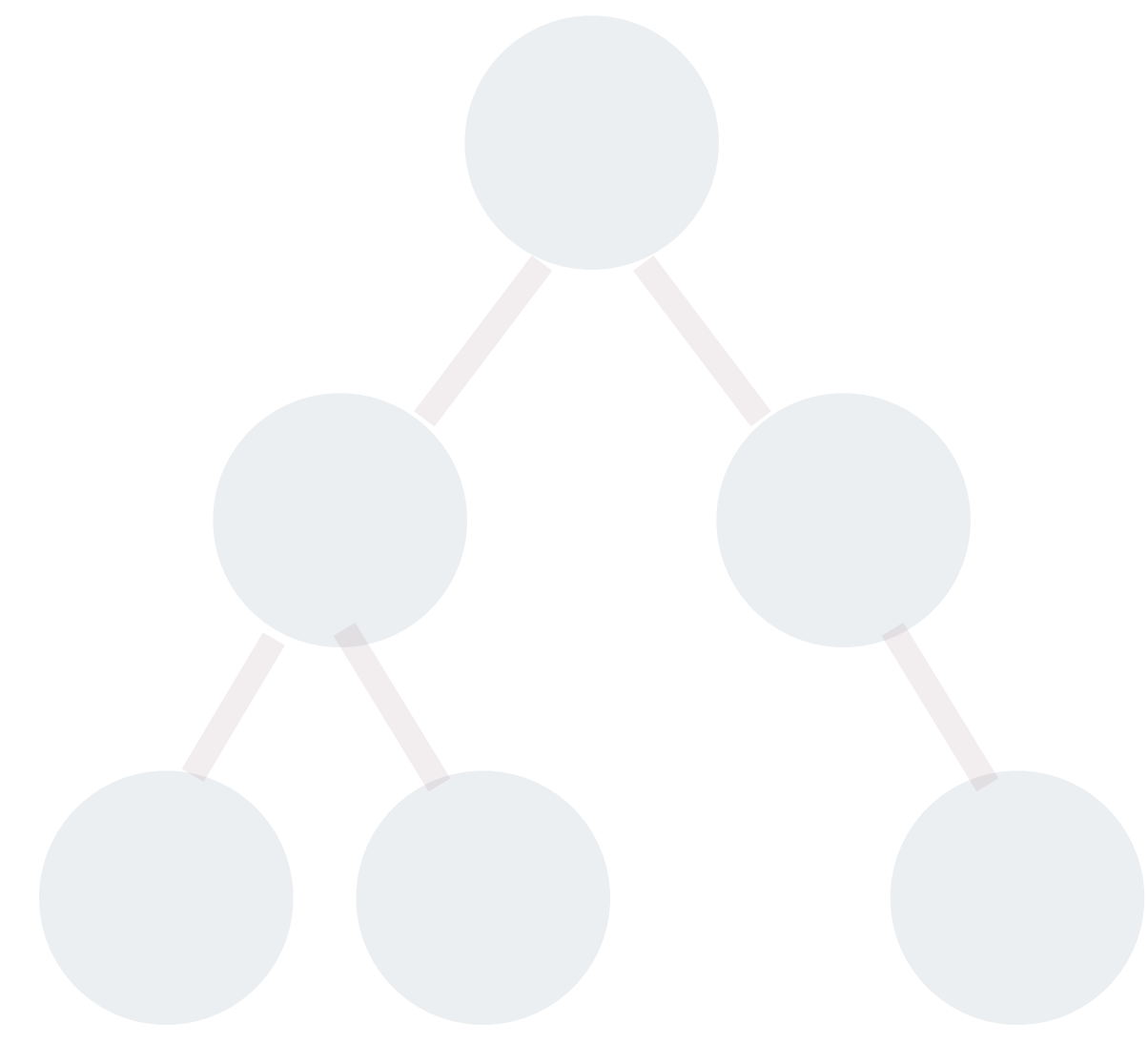
WHY:

- - Breakable
- + Debuggable
- + Performant



WHY:

- - Breakable
- + Debuggable
- + Performant



FULL RECONCILIATION

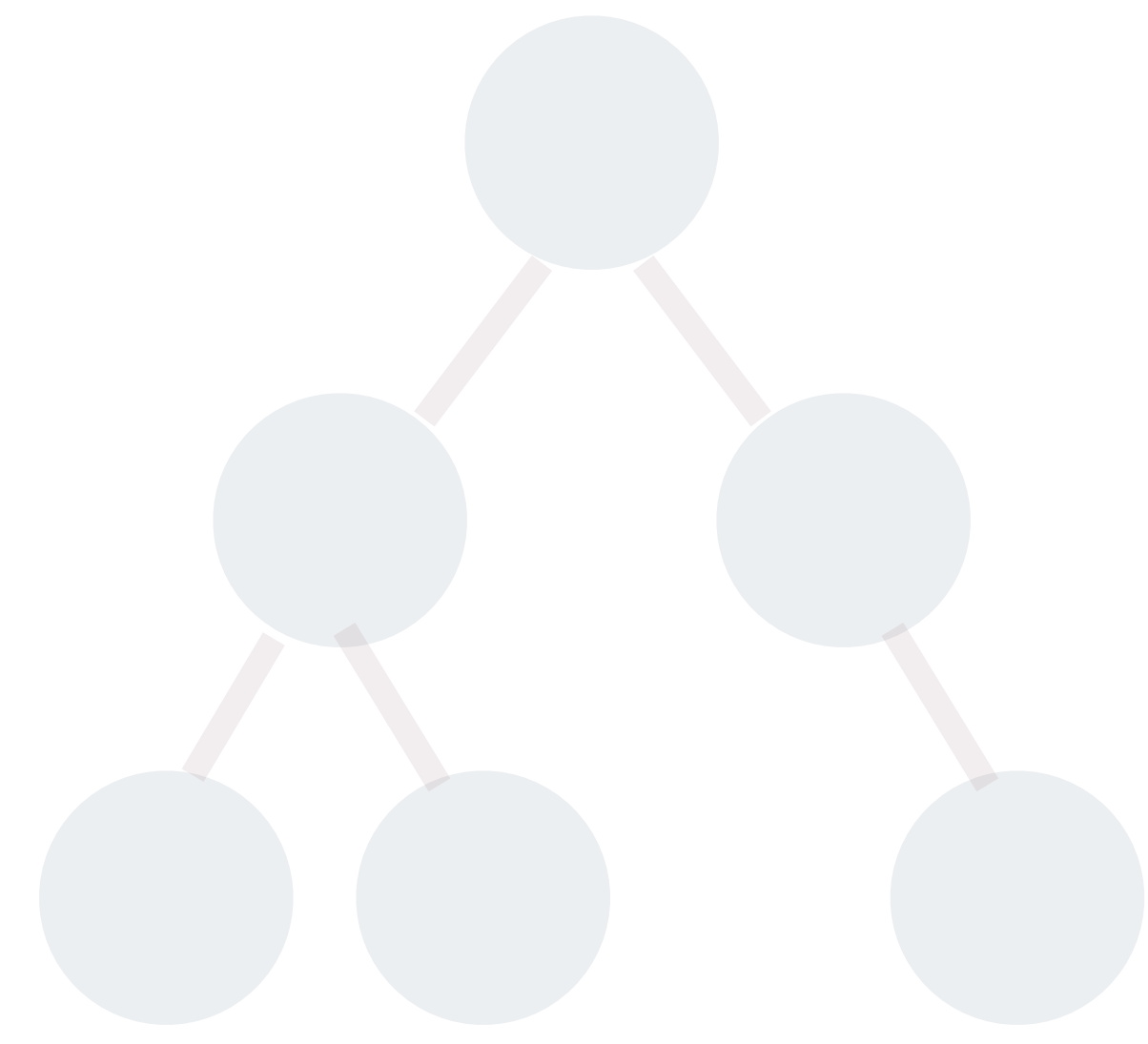
**REACT
RECONCILIATION**

REACT + IMMUTABLE



WHY:

- - Breakable
- + Debuggable
- + Performant



FULL RECONCILIATION
 $O(N^3)$

REACT
RECONCILIATION $O(N)$

REACT + IMMUTABLE
 $O(\log N)$



The screenshot shows the GitHub repository page for `immutable-js/immutable-js`. The repository is described as "Immutable persistent data collections for Javascript which increase efficiency and simplicity." It has 1,555 commits, 4 branches, 65 releases, and 188 contributors. The license is MIT. The current branch is `master`. A list of recent commits is shown, including:

- ThomasdenH and leebyron: Replace any by unknown in typescript types (#1686) - Latest commit 751082f on Feb 14
- `.github`: Update CODE_OF_CONDUCT.md (#1566) - 7 months ago
- `__tests__`: Reapply of 9ea8774 with a Co-authored-by metadata. - 6 months ago
- `contrib/cursor`: Fixing some minor typos (#1439) - 11 months ago
- `pages`: Replace any by unknown in typescript types (#1686) - 2 months ago
- `perf`: Improve performance of toJS (#1581) - 7 months ago
- `resources`: Add files via upload - 4 months ago
- `src`: Improve hash speed and avoid colision for common values (#1629) - 6 months ago

<https://github.com/immutable-js/immutable-js>

Producing a value	size	rest()	map()	has()	Search for va
duce()		butLast()	mapKeys()	includes()	find()
duceRight()	Persistent changes	skip()	mapEntries()	first()	findLast()
ery()	set()	skipLast()	flatMap()	last()	
ne()	delete()	skipWhile()	filter()	Reading deep values	findEntry()
n()	deleteAll()	skipUntil()	flip()	getIn()	findLastEntry
empty()	clear()	take()	filterNot()	hasIn()	findKey()
nt()	update()	takeLast()	reverse()	Conversion to Collections	findLastKey()
ntBy()	merge()	takeWhile()	sortBy()	toMap()	
	mergeWith()	takeUntil()	groupBy()	toOrderedMap()	keyOf()
	mergeDeep()		Conversion to JavaScript types	toSet()	lastKeyOf()
	mergeDeepWith()	Combination	toJS()	toOrderedSet()	
uence algorithms	Deep persistent changes	flatten()	toJSON()	toList()	max()
o()	setIn()		toArray()	toStack()	maxBy()
oKeys()	deleteIn()	Reducing a value	toObject()	Iterators	min()
oEntries()	updateIn()	reduce()	Conversion to Seq	keys()	minBy()
Map()	mergeIn()	reduceRight()	toSeq()	values()	
r()	mergeDeepIn()	every()	toKeyedSeq()	entries()	
)		some()	toIndexedSeq()	Collections (Seq)	
rNot()	Transient changes	join()	toSetSeq()	keySeq()	Comparison
erse()	withMutations()	isEmpty()	concat()	valueSeq()	isSubset()
()	asMutable()	count()	[Symbol.iterator]()	entrySeq()	
o()	wasAltered()	countBy()	Value equality	Side effects	isSuperset()
o()			equals()		

kolodny/immutability-helper: m x +

GitHub, Inc. [US] | https://github.com/kolodny/immutability-helper

Search or jump to... / Pull requests Issues Marketplace Explore

kolodny / immutability-helper

Used by 14,679 Watch 47 Unstar 3,364 Fork 142

Code Issues 4 Pull requests 2 Projects 0 Wiki Security Insights

mutate a copy of data without changing the original source

115 commits 5 branches 28 releases 22 contributors MIT

Branch: master New pull request Create new file Upload files Find File Clone or download

Moshe Kolodny Support set on Maps Latest commit c60fdb8 18 hours ago

__snapshots__	Convert project into TypeScript (#123)	6 months ago
.editorconfig	Convert project into TypeScript (#123)	6 months ago
.gitignore	Convert project into TypeScript (#123)	6 months ago
.npmignore	Prepare for v3.0.0 release	5 months ago
.travis.yml	Support set on Maps	18 hours ago

https://github.com/kolodny/immutability-helper

```
import update from "immutability-helper"
const reducer = (state, action) => {
  switch (action.type) {
    case ADD_TAG:
      return update(state, {
        [action.id]: {
          tags: {
            $push: [action.tag]
          }
        }
      })
    default:
      return state
  }
}
```


A "must-have" for SQL lovers

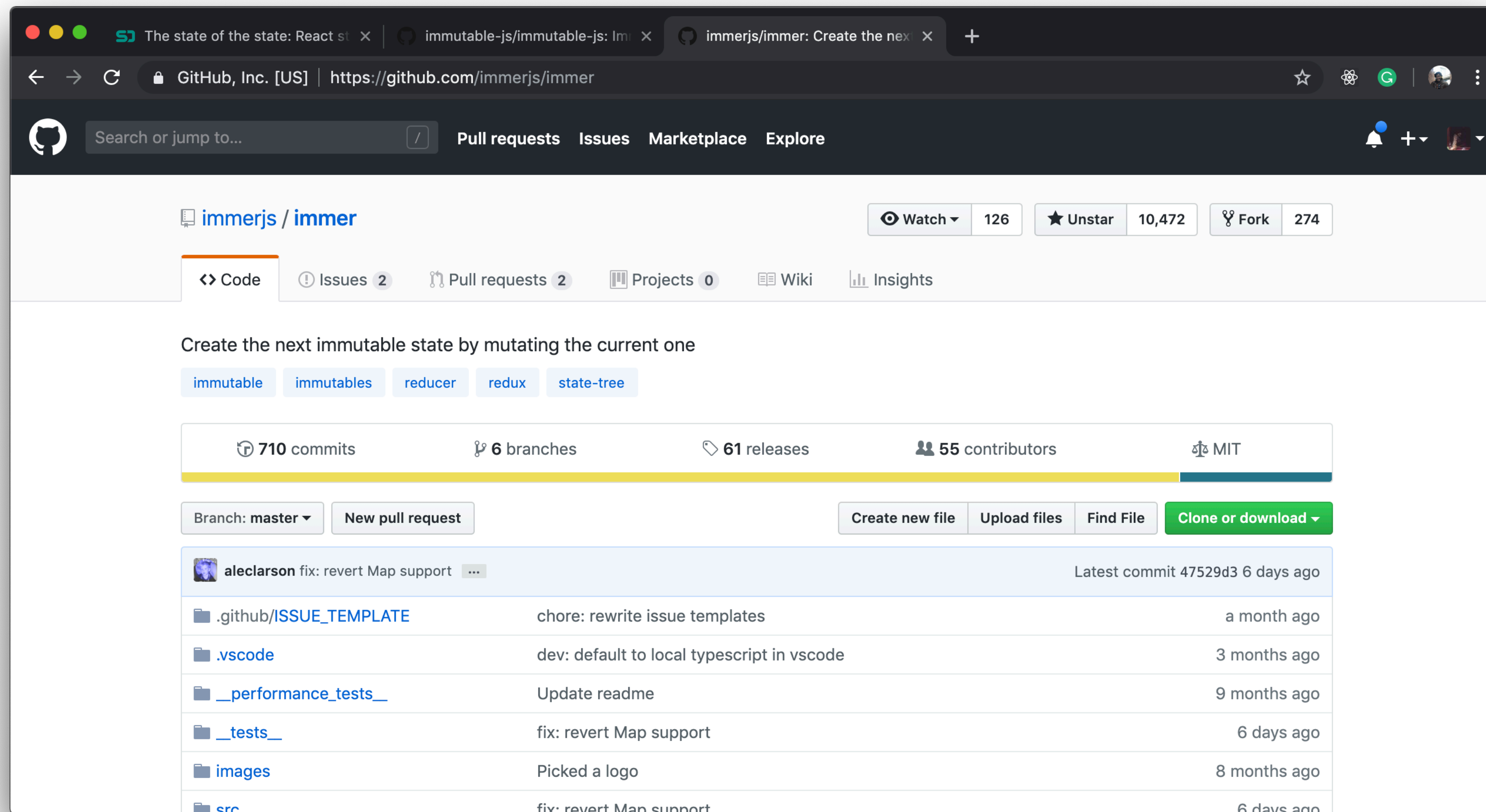


Query languages for client side state

Making simple look hard

O RLY?

Mr. JavaBean



<https://github.com/immerjs/immer>

```
const map1 = { foo: 1, bar: 2 }  
const map2 = Immutable.Map(map1)  
  
const { foo, bar } = map2  
  
console.log(foo) // undefined  
console.log(bar) // undefined
```

```
const map1 = { foo: 1, bar: 2 }  
const map2 = produce(map1, draft => {  
  draft.foo += 10  
})
```

```
const { foo, bar } = map2
```

```
console.log(foo) // 11
```

```
console.log(map1.bar === bar) // true
```

```
const map1 = { foo: 1, bar: 2 }
const map2 = produce(map1, draft => {
  draft.foo += 10
})

const { foo, bar } = map2

console.log(foo) // 11
console.log(map1.bar ≡ bar) // true
```

exposes a single
default function
that does all the
work

```
const map1 = { foo: 1, bar: 2 }
const map2 = produce(map1, draft => {
  draft.foo += 10
})

const { foo, bar } = map2

console.log(foo) // 11
console.log(map1.bar === bar) // true
```

apply all your changes to a temporarily *draft State*, which is a proxy of the *currentState*.

```
const map1 = { foo: 1, bar: 2 }
const map2 = produce(map1, draft => {
  draft.foo += 10
})
```

```
const { foo, bar } = map2
```

```
console.log(foo) // 11
console.log(map1.bar ≡ bar) // true
```

your objects and arrays are really **JavaScript objects and arrays**, so you can do everything you would normally do

```
// add a tag to a todo
{
  ...state,
  [action.id]: {
    ...state[action.id],
    tags: [
      ...state[action.id].tags,
      action.tag
    ]
  }
}

// add a tag to a todo
state[action.id].tags.push(action.tag)
```


ALSO:

- ES5 Fallback
- Strongly typed
- Small (4KB)
- Currying

ALSO:

- Fast Auto freezing
- Equivalent perf to ImmutableJS
- Supports hooks!

If you like MobX, I highly recommend following along [@mweststrate](#)'s work on Immer. While MobX is pretty far removed from the vision of where we're going with React. Immer is dead on.



Sebastian Markbåge @sebmarkbage

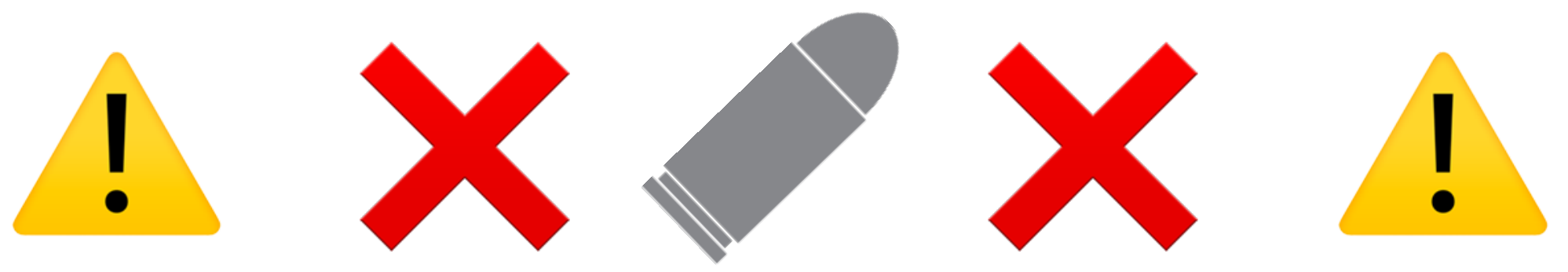
2:43pm - 23 Aug 2018

Recipe for success: take something that's easy to debug, and make it less annoying to write. Thanks to [@mweststrate](#) for immer! [github.com/mweststrate/im...](https://github.com/mweststrate/immer)



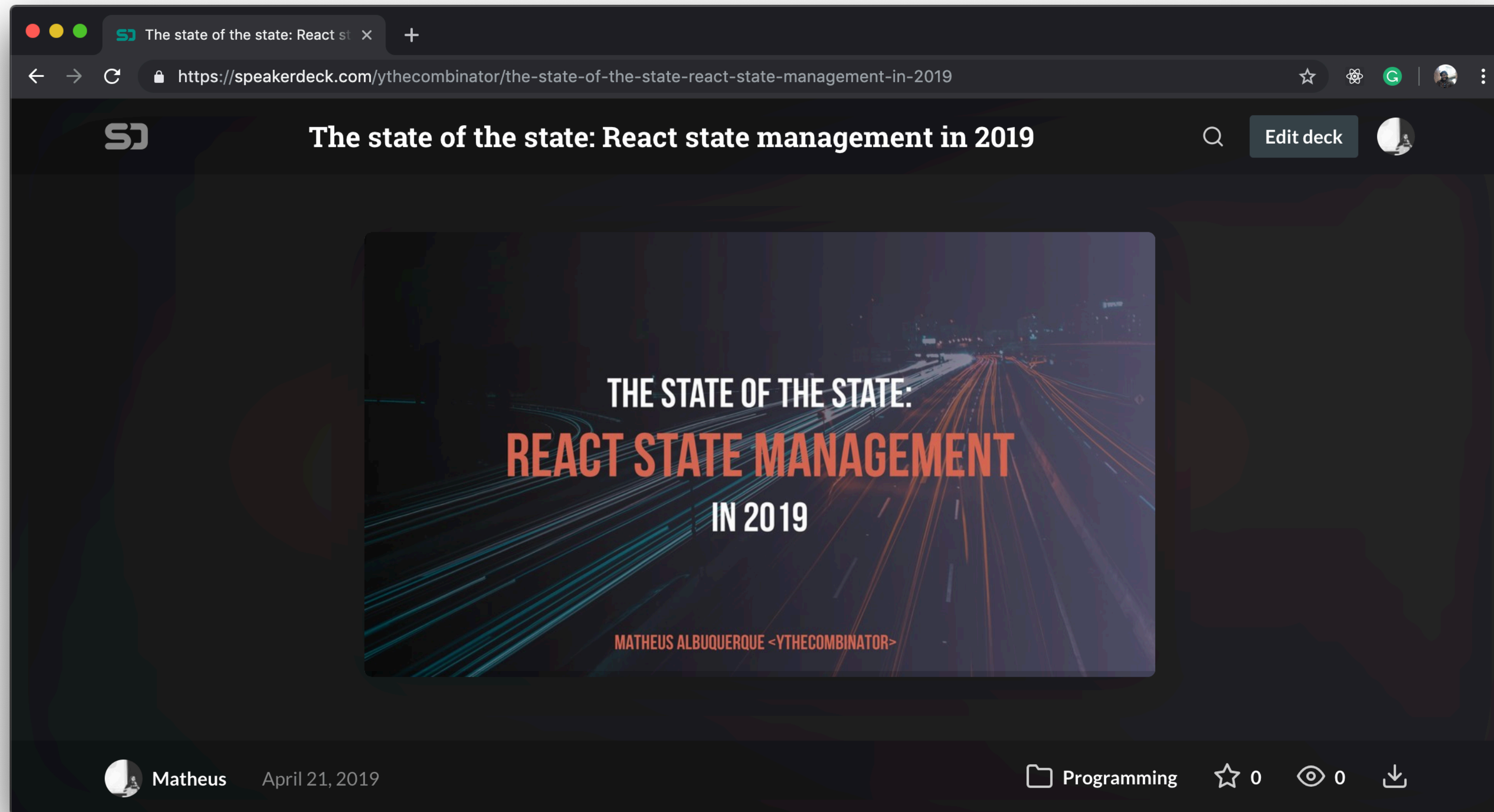
Dan Abramov @dan_abramov

5:50pm - 10 Jul 2018





- ~ senior software engineer, **Front-End**
@beakyn
- ~ **@ythecombinator** on the webs
- ~ addicted to emojis, memes and beer



<https://bit.ly/react-state-management-in-2019>



THANKS!

@YTHECOMBINATOR