

# Who's testing the tests?



- ▀ Mutation testing with StrykerJS

FOSDEM '24

Hi!



*infoSupport*  
Solid Innovator

 I'm Nico

 Manager at Info Support

 Open sourcerer

 Love to swim and run

 nicojs  \_nicojs



## ► So, in this 30 minutes

- Code coverage
- Mutation testing
- StrykerJS
- Open source and consultancy?

... Let's keep it light-hearted tonight 😊





# Code coverage

```
ug'), this.app.set('views',  
this.app.locals.pretty=10, this.setupStaticContent(), this.option  
=await this.readFiles('css'), f==await this.readDirectories:  
path.basename(this.options.cwd), slideSections:  
+ e==await this.listen(); console.log("Present  
); console.log(" Hint: open the url in  
"DRIVE" === t.code? console  
console.error(chalk.r  
)); this.app.list  
>(e));  
ret(t
```

# Code coverage example

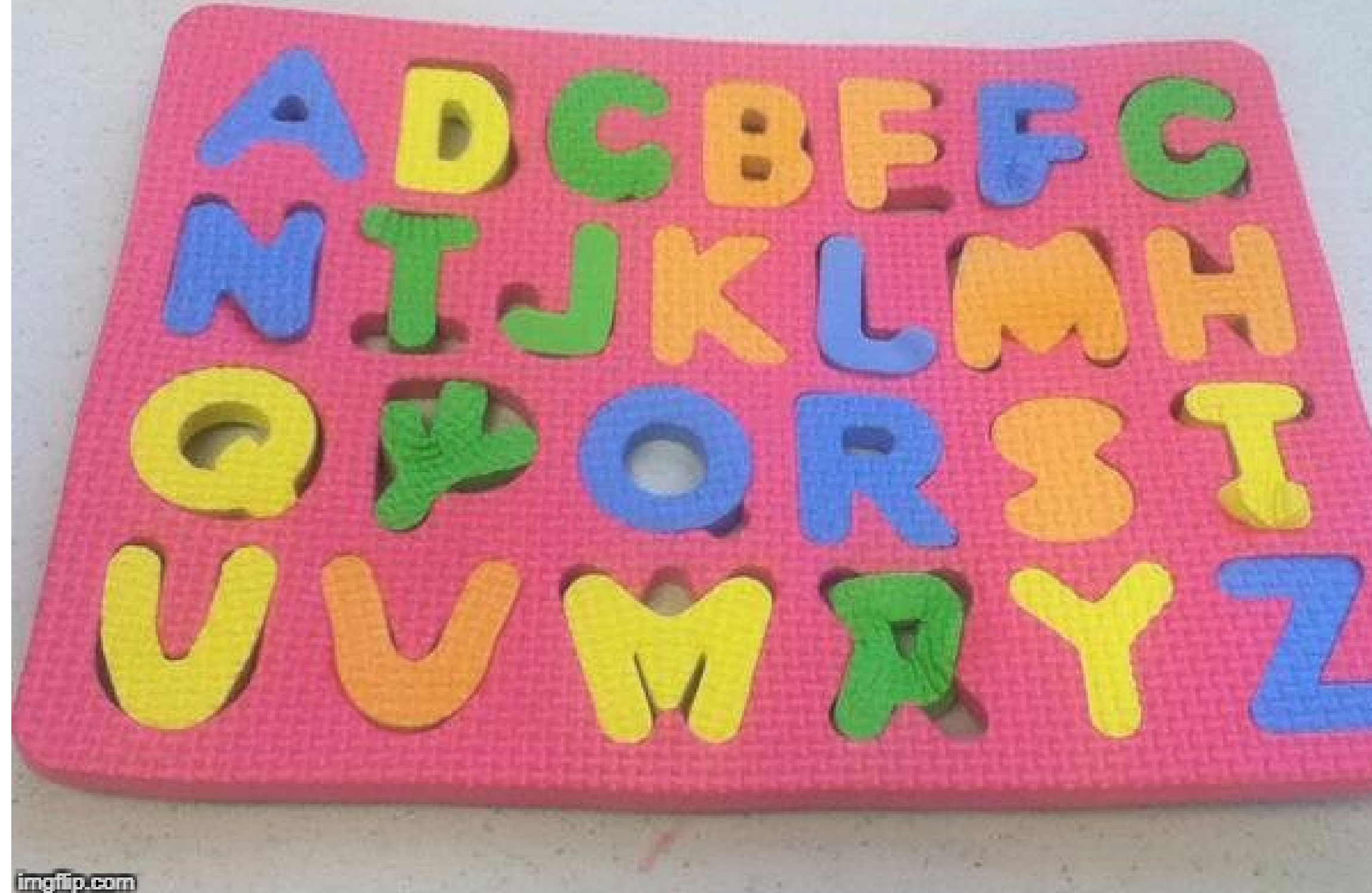
```
1 function isAdult(customer) {  
2   return customer.age >= 18;  
3 }
```

```
1 const customer = { name: 'Professor X', age: 96 };  
2 expect(isAdult(customer)).toEqual(true);
```

What is the code coverage here? 🧦

😞 100% 😞

**ALL UNIT TESTS PASSED!**

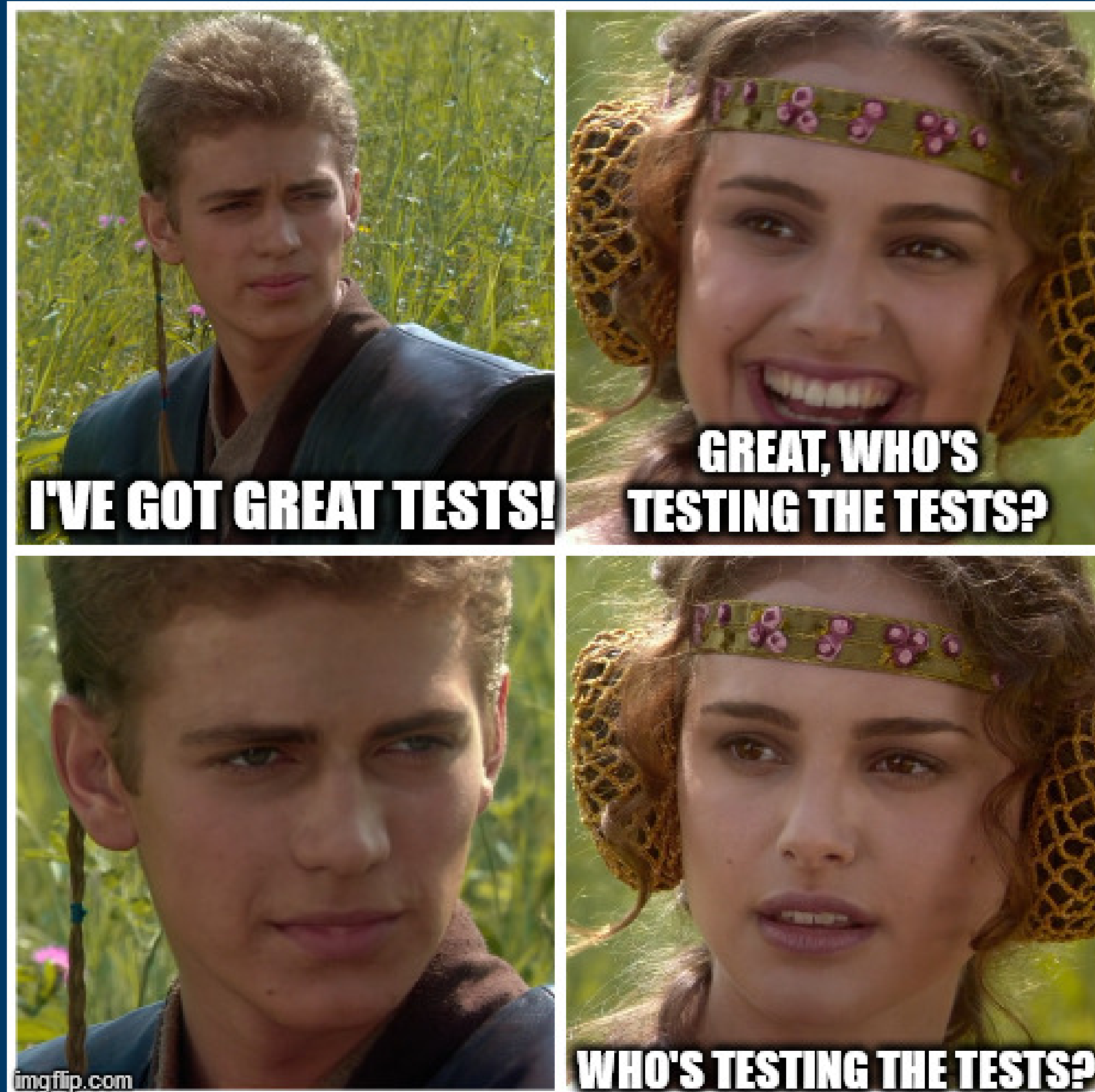




# Mutation testing

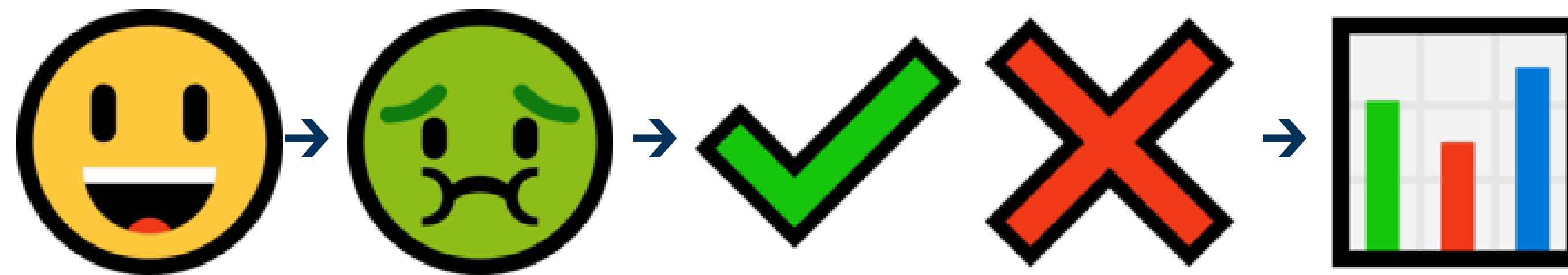
```
ug'), this.app.set('views',  
this.app.locals.pretty=10, this.setupStaticContent(), this.option  
=await this.readFiles('css'), f==await this.readDirectories:  
math.basename(this.options.cwd), slideSections:  
+ e==await this.listen(), console.log("Present  
) console.log(" Hint: open the url in  
"DRINUSE" === t.code? console  
console.error(chalk.r  
) this.app.list  
>(e)))  
st(t
```

# Mutation testing





# Mutation testing process



1. Source code
2. Mutant
3. Killed / survived
4. Report

## ▶ Mutating code

```
function isAdult(customer) {  
  return customer.age >= 18;  
}
```

```
const customer = { name: 'Professor X', age: 96 };  
expect(isAdult(customer)).toEqual(true);
```



# ▶ Mutating code

```
function isAdult(customer) {  
  return customer.age >= 18;  
  /* 🧛 1 */ return customer.age < 18;  
  /* 🧛 2 */ return customer.age > 18;  
  /* 🧛 3 */ return false;  
  /* 🧛 4 */ return true;  
}
```

```
const customer = { name: 'Professor X', age: 96 };  
expect(isAdult(customer)).toEqual(true);
```

🧛 1 Killed / Survived?  
🧛 2 Killed / Survived?  
🧛 3 Killed / Survived?  
🧛 4 Killed / Survived?



# ▶ Mutating code

	1		Killed
	2		Survived
	3		Killed
	4		Survived

How can we improve our result

Add more tests!

- › test when `age < 18`
- › test when `age === 18`

# Common mutations

Original	Mutated
<code>a + b</code>	<code>a - b</code>
<code>a / b</code>	<code>a * b</code>
<code>a &lt; b</code>	<code>a &gt; b</code>
<code>a === b</code>	<code>a !== b</code>
<code>a &amp;&amp; b</code>	<code>a    b</code>
<code>a ?? b</code>	<code>a &amp;&amp; b</code>
<code>const drink = "Cola"</code>	<code>const drink = ""</code>
<code>const list = [1, 2, 3, 4]</code>	<code>const list = []</code>
<code>if (a &gt; b) { ... }</code>	<code>if (true) { ... }</code>
<code>function fn() { ... }</code>	<code>function fn() { /* EMPTY */ }</code>

# Mutation score

```
Killed / #Total * 10
```

A new metric!

- Set a threshold
- Report it from CI
- ...

# ▸ Frameworks

Language	Framework
JavaScript	Stryker
TypeScript	Stryker
Scala	Stryker4s
C#	Stryker.NET
Java	PITest
PHP	InfectionPHP
C/C++	Mull





# StrykerJS






for JS/TS, Vue, Angular, React,  
Svelte, ...



<https://stryker-mutator.io>



# Goals

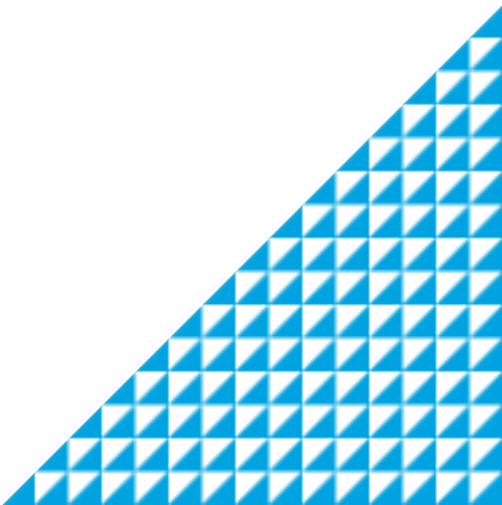
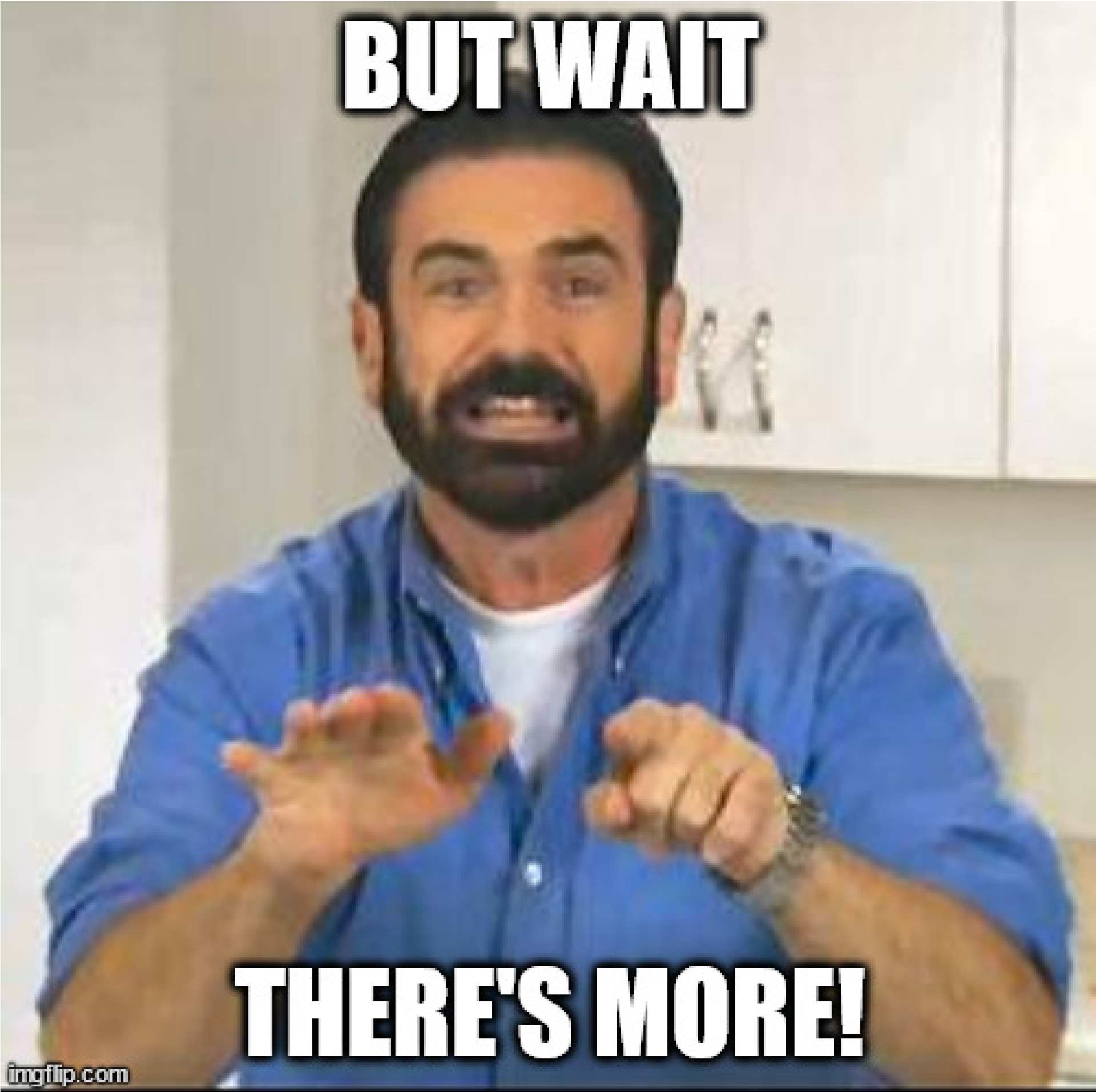
-  Mutation testing for every JS developer.
-  Usability and UX.
-  Ensuring the best performance possible.
-  Plugin support
-  Lowering maintenance

**ALRIGHT HIPPIES**



**SHOW ME  
THE MONEY**





# ▸ StrykerJS features

- Supports 7 test runners out of the box



- Support for **incremental mode**
  - 🖱️ Allowing you to run StrykerJS in your CI pipeline
- Stryker Dashboard for open source projects

MUTATION SCORE **79.4%**

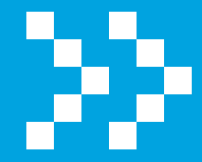
- Test view, another dimension to your mutation testing report

# Weapon regex

Regular expressions are mutated using the awesome [Weapon regex](#) ✕

Original	Mutated
<code>^abc</code>	<code>abc</code>
<code>abc\$</code>	<code>abc</code>
<code>[abc]</code>	<code>[^abc]</code>
<code>\d</code>	<code>\D</code>
<code>\s</code>	<code>\S</code>
<code>\w</code>	<code>\W</code>
<code>a? / a* / a+ / a{1,3}</code>	<code>a</code>





# Open source and consultancy?

```
ug'), this.app.set('views',  
this.app.locals.pretty=10, this.setupStatiContent(), this.option  
=await this.readFiles('css'), f==await this.readDirectories:  
math.basename(this.options.cwd), slideSections:  
+ e==await this.listen(); console.log("Present  
); console.log(" Hint: open the url in  
"DRIVE" === t.code? console  
); console.error(chalk.r  
); this.app.list  
>(e));  
ret(t
```

# Open source

Stryker is an open source initiative

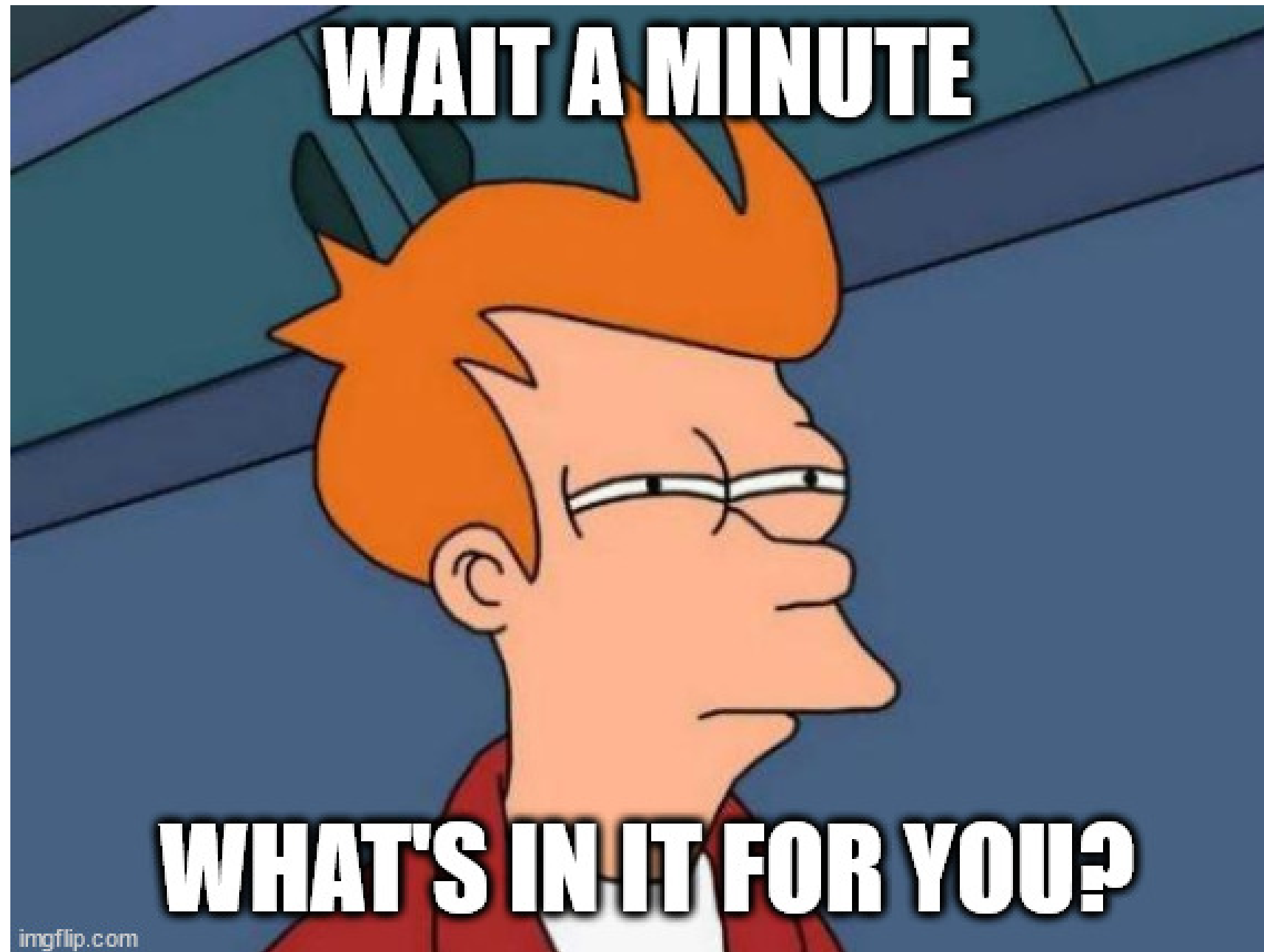
## ➤ The three flavors

- **StrykerJS**
- **Stryker.NET** -- for C#
- **Stryker4s** -- for Scala

## ➤ Side projects

- **Mutation testing elements**  
Custom elements designed to display a mutation testing report.
- **Weapon Regex**  
Mutate your regular expressions in Scala and JS
- **Stryker Dashboard**  
Host your mutation testing reports in a website
- **Calisto**  
Calculate the quality of mutation operators

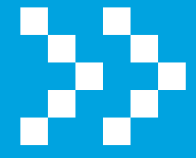
# Who is thinking? ... be honest 🧦



- **Talent acquisition**  
Student tracks, Hackathons, Graduation assignments
- **Thought leadership**  
Bad tests? Who you gonna call?
- **Social Responsibility**  
*We use, but also give back*







# Wrap-up

```
ug'), this.app.set('views',  
this.app.locals.pretty=10, this.setupStaticContent(), this.option  
=await this.readFiles('css'), f==await this.readDirectories:  
path.basename(this.options.cwd), slideSections:  
t e==await this.listen(); console.log("Present  
); console.log(" Hint: open the url in  
"DRIVEUSE" === t.code? console  
); console.error(chalk.r  
); this.app.list  
>(e));  
ret(t
```

# Wrap-up

- Code coverage **doesn't** test *effectiveness*.
- Mutation testing **does** test effectiveness.
- **StrykerJS**, mutation testing for every JS developer
- **Open source** can be a good fit for a consultancy company

# More mutation testing?

## How Mutation Testing Got Practical

Sunday, 14:45-15:15

Testing & CD devroom, UD2.208 (Decroly)

- Deep dive
- Performance enhancements
- State-of-the-art innovations

# Thank you!

[github.com/nicojs/presentations](https://github.com/nicojs/presentations)



[stryker-mutator.io](https://stryker-mutator.io)

**infoSupport**  
Solid Innovator

**Nico Jansen**

Manager and open source enthusiast.

 nicojs  \_nicojs

[opensource.infosupport.com](https://opensource.infosupport.com)