Who's testing the tests?

Mutation testing with StrykerJS

FOSDEM '24
Hi!

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
Code coverage example

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

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

What is the code coverage here? 🙋

😢 100% 😢
ALL UNIT TESTS PASSED!
Mutation testing
Mutation testing

I'VE GOT GREAT TESTS!

GREAT, WHO'S TESTING THE TESTS?

WHO'S TESTING THE TESTS?
Mutation testing process

1. Source code
2. Mutant
3. Killed / survived
4. Report
function isAdult(customer) {
    return customer.age >= 18;
}

const customer = { name: 'Professor X', age: 96 };
extpect(isAdult(customer)).to.equal(true);
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)).to.equal(true);
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

<table>
<thead>
<tr>
<th>Original</th>
<th>Mutated</th>
</tr>
</thead>
<tbody>
<tr>
<td>a + b</td>
<td>a - b</td>
</tr>
<tr>
<td>a / b</td>
<td>a * b</td>
</tr>
<tr>
<td>a &lt; b</td>
<td>a &gt; b</td>
</tr>
<tr>
<td>a === b</td>
<td>a !== b</td>
</tr>
<tr>
<td>a &amp;&amp; b</td>
<td>a</td>
</tr>
<tr>
<td>a ?? b</td>
<td>a &amp;&amp; b</td>
</tr>
<tr>
<td>const drink = &quot;Cola&quot;</td>
<td>const drink = &quot;&quot;</td>
</tr>
<tr>
<td>const list = [1, 2, 3, 4]</td>
<td>const list = []</td>
</tr>
<tr>
<td>if (a &gt; b) { ... }</td>
<td>if (true) { ... }</td>
</tr>
<tr>
<td>function fn() { ... }</td>
<td>function fn() { /* EMPTY */ */</td>
</tr>
</tbody>
</table>
Mutation score

\[
\text{Killed / Total} \times 10
\]

A new metric!

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

(*) Note: The exact formula may vary depending on the context. Always refer to the official documentation or guidelines for precise calculation.
# Frameworks

<table>
<thead>
<tr>
<th>Language</th>
<th>Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>JavaScript</td>
<td>Stryker</td>
</tr>
<tr>
<td>TypeScript</td>
<td>Stryker</td>
</tr>
<tr>
<td>Scala</td>
<td>Stryker4s</td>
</tr>
<tr>
<td>C#</td>
<td>Stryker.NET</td>
</tr>
<tr>
<td>Java</td>
<td>PITest</td>
</tr>
<tr>
<td>PHP</td>
<td>InfectionPHP</td>
</tr>
<tr>
<td>C/C++</td>
<td>Mull</td>
</tr>
</tbody>
</table>
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
BUT WAIT

THERE'S MORE!
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
- Test view, another dimension to your mutation testing report
## Weapon regex

Regular expressions are mutated using the awesome Weapon regex

<table>
<thead>
<tr>
<th>Original</th>
<th>Mutated</th>
</tr>
</thead>
<tbody>
<tr>
<td>^abc</td>
<td>abc</td>
</tr>
<tr>
<td>abc$</td>
<td>abc</td>
</tr>
<tr>
<td>[abc]</td>
<td>[^abc]</td>
</tr>
<tr>
<td>\d</td>
<td>\D</td>
</tr>
<tr>
<td>\s</td>
<td>\S</td>
</tr>
<tr>
<td>\w</td>
<td>\W</td>
</tr>
<tr>
<td>a? / a* / a+ / a{1,3}</td>
<td>a</td>
</tr>
</tbody>
</table>
Open source and consultancy?
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
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

Nico Jansen
Manager and open source enthusiast.

github.com/nicojs

nicojs

Opensource.infosupport.com

tryker-mutator.io