



# Testautomatisierung ohne Assertions

Dr. Jeremias Rößler

<https://retest.de>



# Role-Model: Unit-Testing

JUnit

```

@Test
public void extended_xml_should_be_removed_correctly() throws Exception {
    // Prepare state
    final String xml = "<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n" //
        + "<root>\n" //
        + "\t<parent>\n" //
        + "\t\t<remove>should be removed</remove>\n" //
        + "\t\t<keep>should not be removed</keep>\n" //
        + "\t</parent>\n" //
        + "</root>";
    final RemoveElementTransformer transformer = new RemoveElementTransformer( "parent", "remove" );

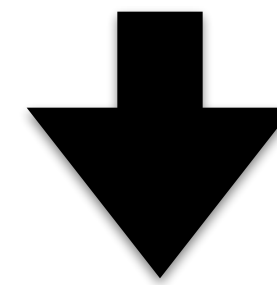
    // Execute feature under test
    final InputStream transform = transformer.transform( new ByteArrayInputStream( xml.getBytes() ) );

    // Assert result
    final String result = IOUtils.toString( transform, "UTF-8" );
    Assert.assertFalse( result.contains( "remove" ) );
    Assert.assertTrue( result.contains( "keep" ) );
}


```



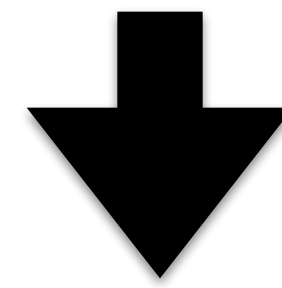
```
<?xml version="1.0" encoding="UTF-8" ?>  
<root>  
  <parent>  
    <remove>should be removed</remove>  
    <keep>should not be removed</keep>  
  </parent>  
</root>
```



```
<?xml version=\"1.0\" encoding=\"UTF-8\" ?>  
<root>  
  <parent>  
    <keep>should not be removed</keep>  
  </parent>  
</root>
```



```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <parent>
    <remove>should be removed</remove>
    <keep>should not be removed</keep>
  </parent>
</root>
```

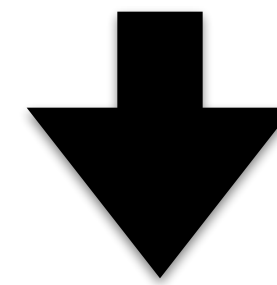


```
<?xml keep</keep>
  </parent>
</root>
```

✓ passes



```
<?xml version="1.0" encoding="UTF-8" ?>  
<root>  
  <parent>  
    <remove>should be removed</remove>  
    <keep>should not be removed</keep>  
  </parent>  
</root>
```



```
<?xml version=\"1.0\" encoding=\"UTF-8\" ?>  
<root>  
  <parent>  
    <keep>should not be removed</keep>  
  </parent>  
</root>
```



Capitol

THE BEATLES  
HEY JUDE

1

C1-50442

© 1968 CAPITOL RECORDS, INC.  
ALL RIGHTS RESERVED  
REPRODUCED BY CAPITOL RECORDS, INC.  
MADE IN U.S.A.  
C1-50442-A  
C1-50442-B



## Approval Tests

```
[TestMethod]
public void TestPurchase()
{
    // Create a new receipt
    var r = new Receipt();
    // Add 1 candy bar at $.50
    r.AddItem(1, "Candy Bar", 0.50);
    // Add 2 sodas bar at $1
    r.AddItem(2, "Soda", 1.0);
    // verify the receipt
    Approvals.Verify(r.ToString());
}
```


ReceiptTest.TestPurchase.receive		ReceiptTest.TestPurchase.app	
1		1	
2	1 · Candy · Bar · @ · \$0.50	2	1 · Candy · Bar · @ · \$0.50
3	2 · Soda · @ · \$1.00 = \$2.00	3	2 · Soda · @ · \$1.00 = \$2.00
4	Subtotal = \$2.50	4	Total = \$2.50
5	Tax (10%) = \$0.25		
6	Total = \$2.75		



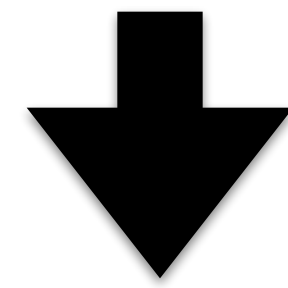
```

@Test
public void extended_xml_should_be_removed_correctly_Approvals() throws Exception {
    // Prepare state
    final String xml = "<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n" //
        + "<root>\n" //
        + "\t<parent>\n" //
        + "\t\t<remove>should be removed</remove>\n" //
        + "\t\t<keep>should not be removed</keep>\n" //
        + "\t</parent>\n" //
        + "</root>";
    final RemoveElementTransformer transformer = new RemoveElementTransformer( "parent", "remove" );
    final InputStream transform = transformer.transform( new ByteArrayInputStream( xml.getBytes() ) );
    ApprovalsUtil.verifyXml( IOUtils.toString( transform, "UTF-8" ) );
}

```



```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <parent>
    <remove>should be removed</remove>
    <keep>should not be removed</keep>
  </parent>
</root>
```



```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <parent>
    <keep>should not be removed</keep>
  </parent>
</root>
```



**Whitelisting**  
of changes



Ideal amount  
of checks

**Blacklisting**  
of changes

retest.org

1

## Multiple choice:



100 characters remaining ?

Randomize answer order for each respondent ⇅

Randomize answer order for each respondent ?

enter answer text

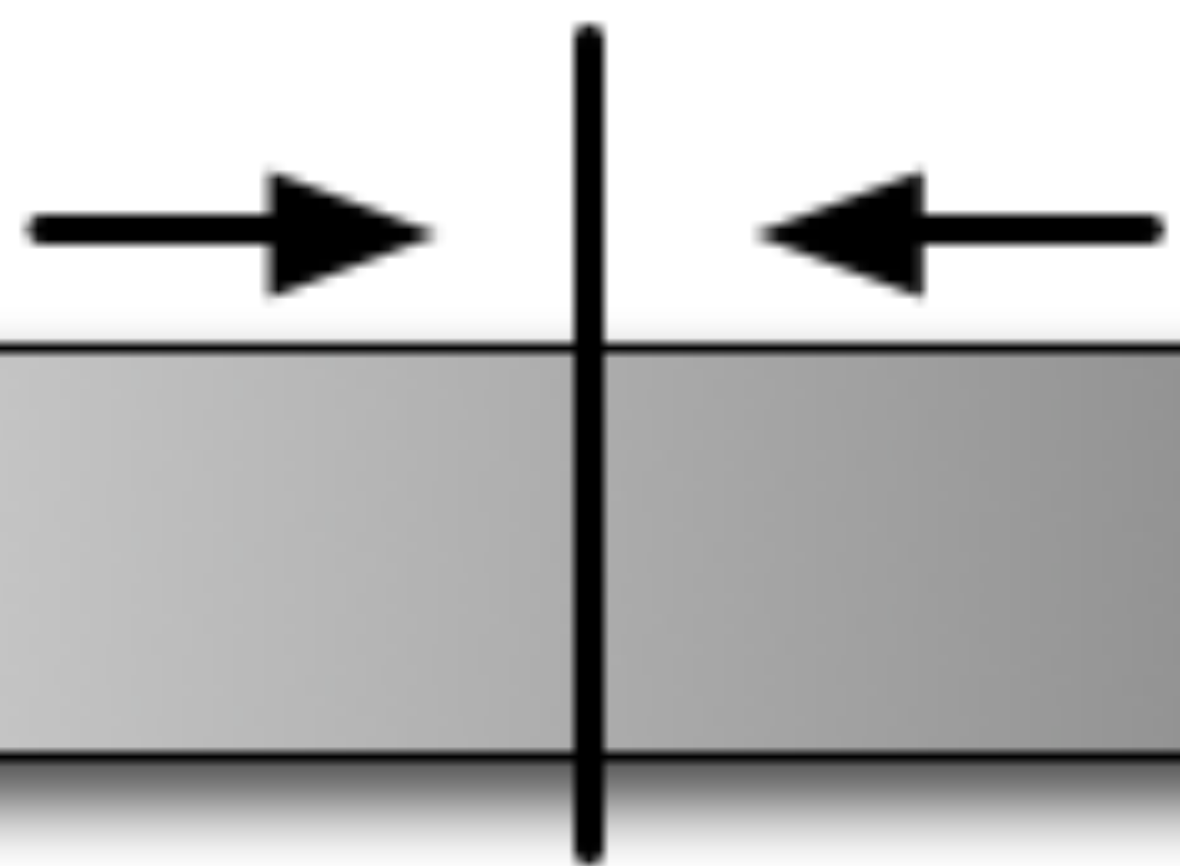




**Whitelisting  
of changes**

check  
everything

Ideal amount  
of checks



**Blacklisting  
of changes**

check  
nothing

retest.org



retest.org

There should be a form here:



Here's a checkbox:

- Cheese
- Peas
- Cheese and peas
- Not a sausage
- Not another sausage

I like cheese


Cumberland sausage

Label



# Live Demo: SW Changes



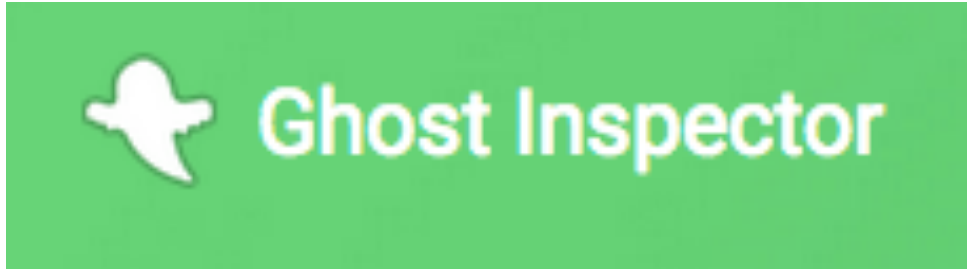
 [wearefriday / spectre](#)


CSS Critic



WRAITH

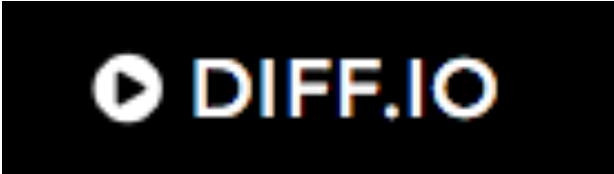
 [gemini-testing / gemini](#)



 Pix-Diff




PhantomCSS



 applitools

backTrac

 [garris / BackstopJS](#)



 [rprt / rspec-page-regression](#)

automated-screenshot-diff



 [bslatkin / dpxdt](#)

Kobold

 Visual Review





Define a new tested application

Full name of application:  
PyGTK Bug Tracker

File extension to use for TextTest files associated with this application:  
bugs

Subdirectory name to store the above application files under (leave blank for local storage):  
ex4\_uitest

Java Class name (instead of executable program):  
PyGTK GUI with StoryText

Enable CaptureMock for record/playback mocking

Select executable program to test:  
carm proj textttest geoff course tests **ex4\_uitest**

Location: bugsystem.py

Name	Size	Modified
bugsystem.py	4.9 KB	11:13

Cancel OK



# Jest

 Delightful JavaScript Testing

TRY OUT JEST

GET STARTED

WATCH TALKS

LEARN MORE

★ Star 18,072



## Developer Ready

Complete and ready to set-up JavaScript testing solution. Works out of the box for any React project.



## Instant Feedback

Fast interactive watch mode runs only test files related to changed files and is optimized to give signal quickly.



## Snapshot Testing

Capture snapshots of React trees or other serializable values to simplify testing and to analyze how state changes over time.



automated  
regression test  
≠  
test



automated  
regression test  
=  
version control



file:///home/joshua/dev/cpp/cps111/bigcalc/apstring.cpp | file:///home/joshua/dev/cpp/cps111/bigcalc/apstring.h

File Difference Settings Help

Viewing: **apstring.cpp** Previous File Next File Files

```

/home/joshua/dev/cpp/cps111/bigcalc/apstring.cpp
234 apstring operator + ( char ch, const apstring & str ) const {
235 // precondition: returns concatenation of ch & str
236 {
237     apstring result; // make string equivalent
238     result = ch;
239     result += str;
240     return result;
241 }
242
243 apstring operator + ( const apstring & str, char ch ) const {
244 // precondition: returns concatenation of str & ch
245 {
246     apstring result(str);
247     result += ch;
248     return result;
249 }
250
251
252 apstring apstring::substr(int pos, int len) const {
253 //description: extract and return the substring
254 //              at index pos
255 //precondition: this string represents c0, c1, ..., cn-1
256 //              0 <= pos <= pos + len - 1 < n.

```

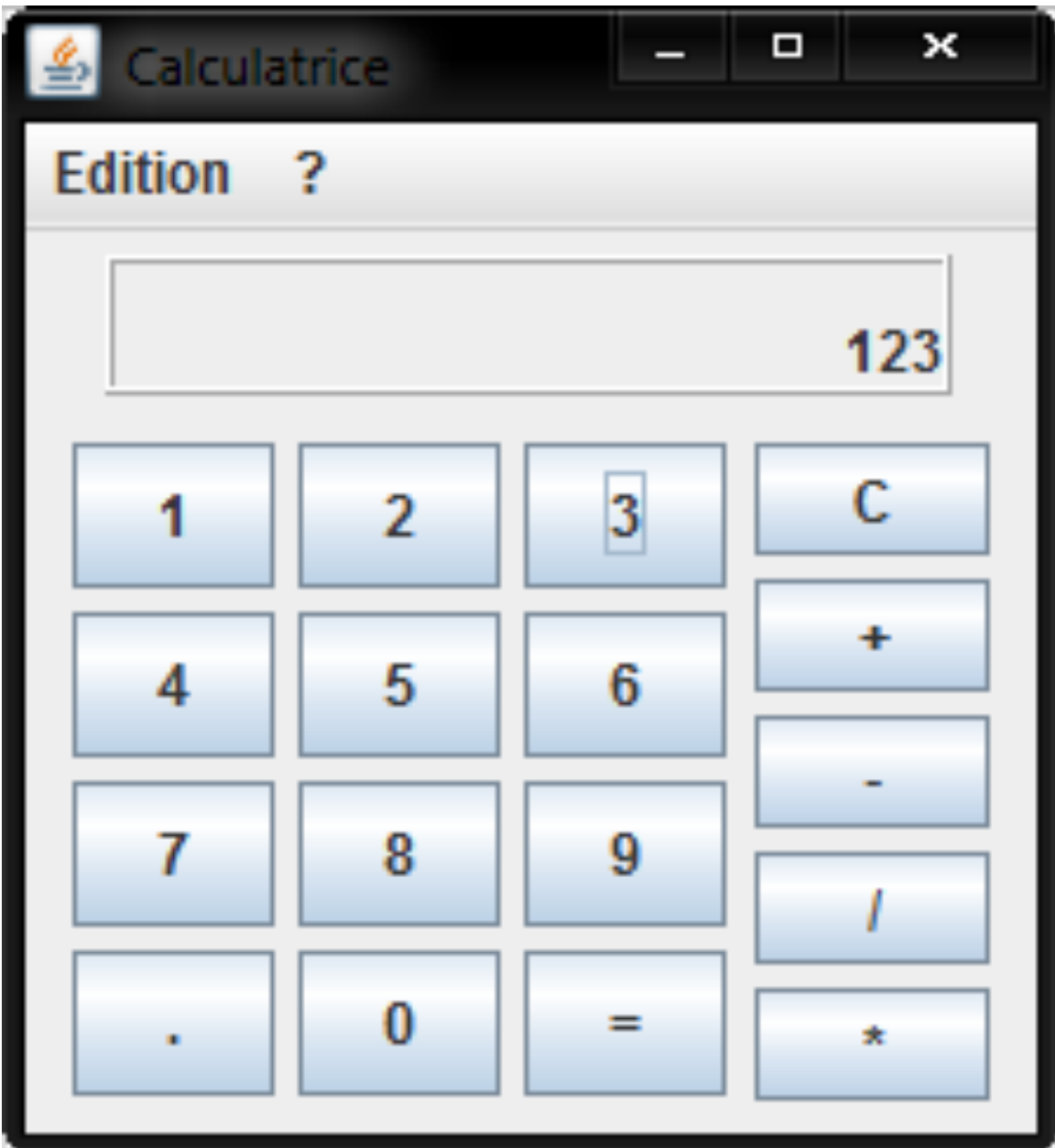
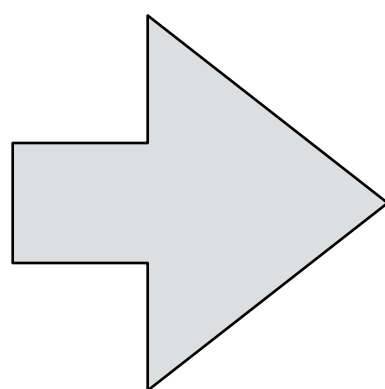
```

/home/joshua/dev/cpp/cps111/bigcalc/apstring.h
43
44 // indexing
45
46 char operator[] ( int k ) const;
47 char & operator[] ( int k );
48
49 // modifiers
50
51 const apstring & operator += ( const apstring & str );
52 const apstring & operator += ( char ch );
53
54
55 private:
56     int myLength; // length
57     int myCapacity; // capacity
58     char * myCString; // storage
59 };
60
61 // The following free (non-member) functions are defined in
62 //
63 // I/O functions
64

```



```
function date_mysql2ger($datex) {  
    list($year, $month, $day) = explode("-", $datex);  
    return sprintf("%02d.%02d.%04d", $day, $month, $year);  
}  
  
function date_ger2mysql($datex) {  
    list($day, $month, $year) = explode("-", $datex);  
    return sprintf("%04d-%02d-%02d", $year, $month, $day);  
}  
  
function timestamp_mysql2ger($t) {  
    return sprintf("%02d.%02d.%04d",  
        substr($t, 6, 2),  
        substr($t, 4, 2),  
        substr($t, 0, 4));  
}  
  
function date_mysql2ger($datex) {  
    list($year, $month, $day) = ex  
    return sprin
```





# Close the Gap: *automated* Tests



```
private WebDriver driver;
```

```
@Before
```

```
public void setup() {  
    System.setProperty("webdriver.chrome.driver", "src/test/resources/chromedriver");  
    driver = new ChromeDriver();  
    driver.manage().timeouts().pageLoadTimeout(-1, TimeUnit.MINUTES);  
}
```

```
@Test
```

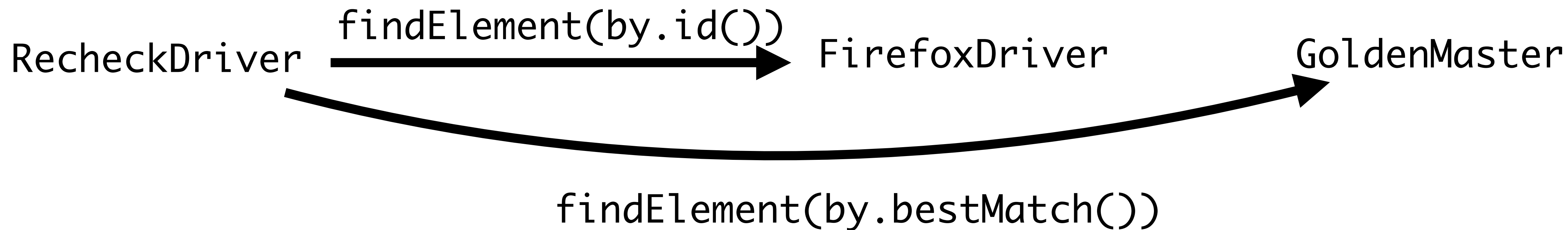
```
public void check_add() throws Exception {  
    driver.get("http://localhost:8080/login");  
    // login  
    driver.findElement(By.name("username")).sendKeys("Max");  
    driver.findElement(By.name("password")).sendKeys("retest");  
    driver.findElement(By.name("login")).click();  
  
    // test  
    driver.findElement(By.id("1")).click();  
    driver.findElement(By.id("plus")).click();  
    driver.findElement(By.id("1")).click();  
    driver.findElement(By.id("equals")).click();  
  
    Assert.assertEquals(driver.findElement(By.name("result")).getAttribute("value"), "2.0");  
}
```

```
@After
```

```
public void tearDown() {  
    driver.quit();  
}
```



# Live Demo: Breaking Test



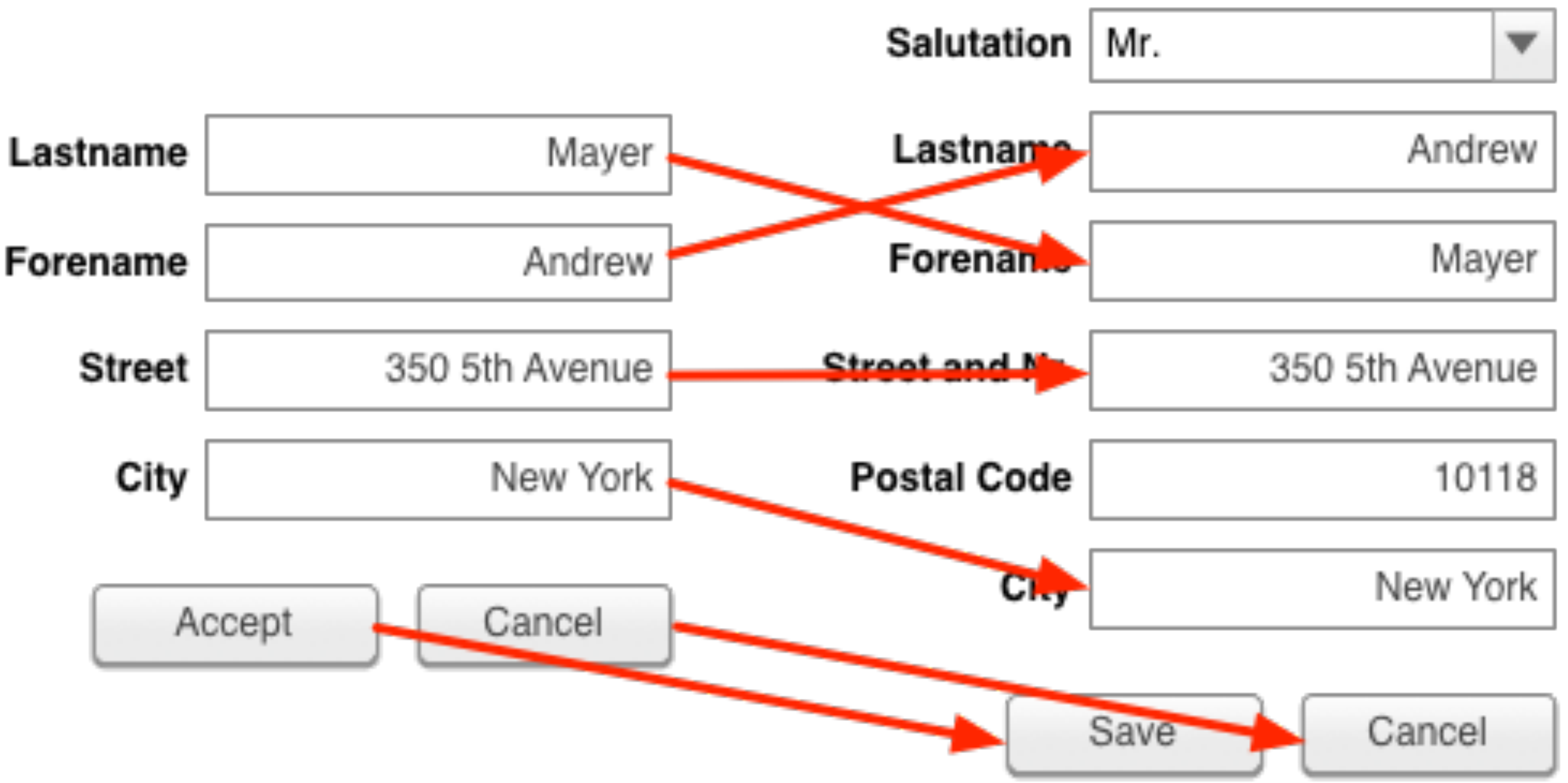
```
***** recheck warning *****  
The HTML id attribute used for element identification changed from 'intro-slider' to 'introSlider'.  
retest identified the element based on the persisted old state.  
If you apply these changes to the state , your test will break.  
Use `By.id("introSlider")` or `By.retestId("9c40281d-5655-4ffa-9c6d-d079e01bb5a3")` to update your test.
```

# Multilocators

Path: `JPanel_0/JPanel_0/JTextField_0` Name: `login`

Label: `Username`

Pixel coordinates: `(71, 1)`





# **Live Demo: Unbreakable Changes**

Reason number 34986 why I hate selenium. People write code like this and are proud of it: result = self.driver.find\_element\_by\_css\_selector('#root > div > ul > li > div > p')

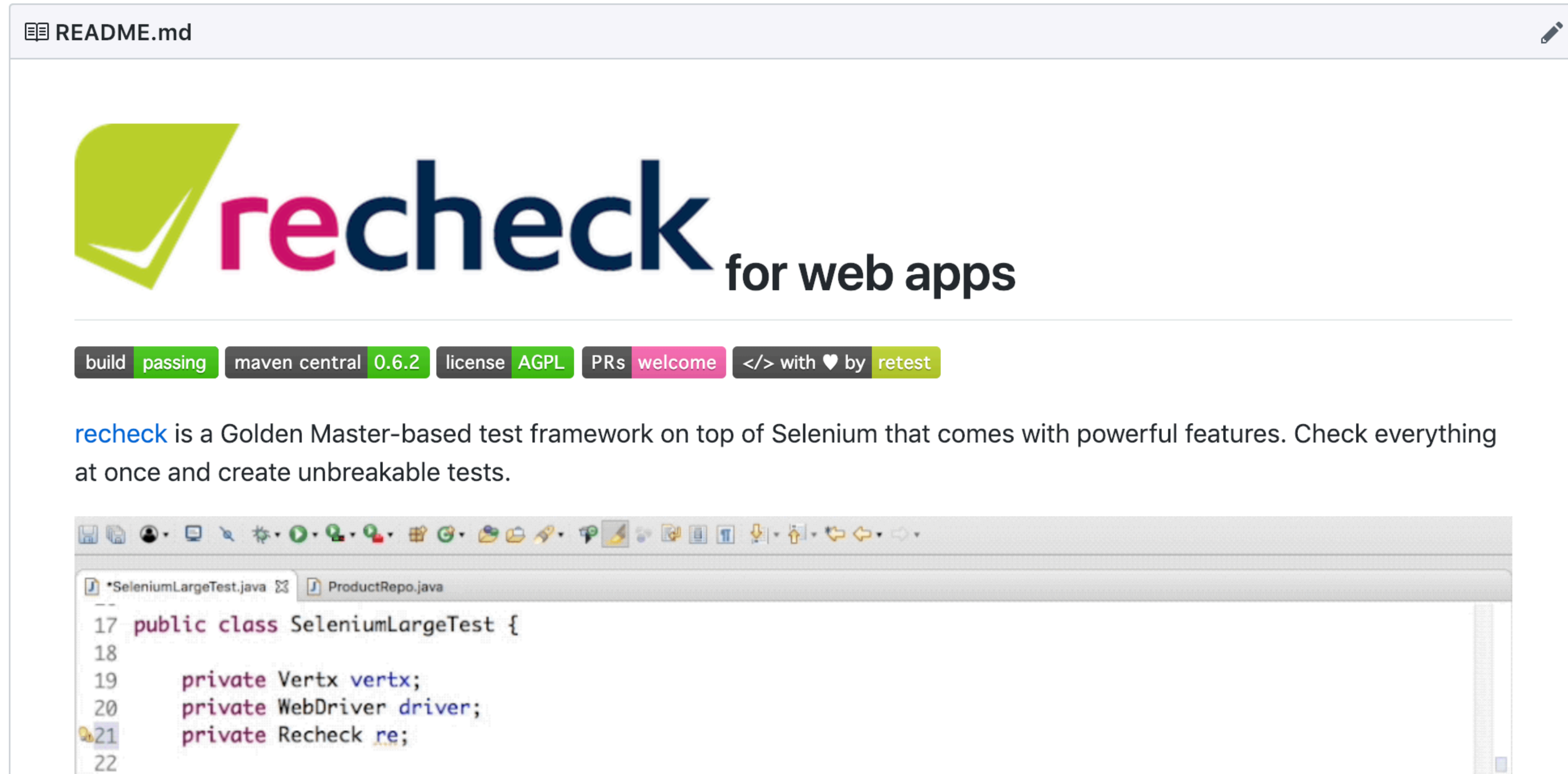


**Mr. Weasel** @alanpage

1:06am - 16 May 2019

<https://twitter.com/alanpage/status/1128798933306597377>

<https://github.com/retest/recheck-web>



README.md

# recheck for web apps

build passing maven central 0.6.2 license AGPL PRs welcome </> with ❤️ by retest

**recheck** is a Golden Master-based test framework on top of Selenium that comes with powerful features. Check everything at once and create unbreakable tests.

```
*SeleniumLargeTest.java ProductRepo.java
17 public class SeleniumLargeTest {
18
19     private Vertx vertx;
20     private WebDriver driver;
21     private Recheck re;
22
```



# Creating Unbreakable Selenium Tests

## Bitte geben Sie uns jetzt Ihr Feedback!

Testautomatisierung ohne Assertions

*Dr. Jeremias Rößler*



[@roessler](https://twitter.com/roessler)



[roessler@retest.de](mailto:roessler@retest.de)



### Nächste Vorträge in diesem Raum

**14:30** TDD demystified, *Tilmann Glaser,*  
*Peter Fichtner*

**15:45** Legacy Code im Griff dank Mikado  
Methode, *Falk Sippach*

