Gmail

Philip Riecks <mail@philipriecks.de>

[Test] Stop Writing Unit Tests ❌

Philip (rieckpil) <blog@rieckpil.de> 3. März 2023 um 16:27
Antwort an: us20-02ddfee17b-18f19694a8@inbound.mailchimp.com
An: fellow subscriber <mail@philipriecks.de>

Stop Writing Unit Tests and Do More Interesting Stuff

Hi fellow subscriber,

There are certain things you're unlikely to hear from a Java developer. Like: *"I'd love to spend lots more of my time writing unit tests!"*

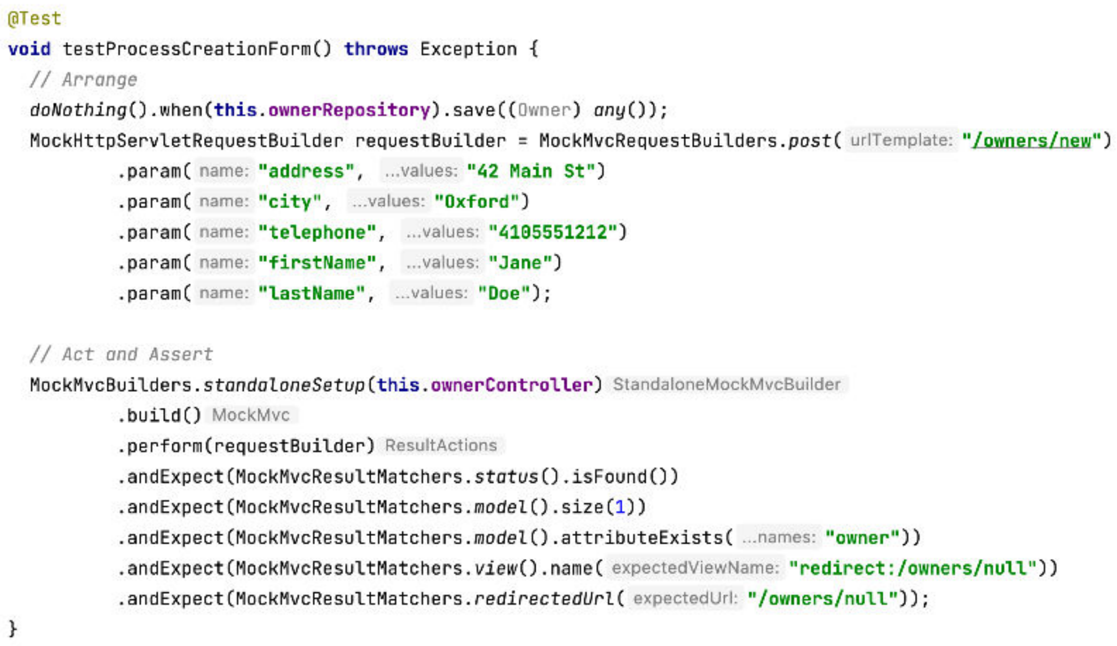
The opposite is usually true. Even though we all know unit testing is important, maybe even critical, most of us see writing the actual tests as a pretty boring, repetitive task.

Which it often is, no matter how critical. And there are so many more interesting problems just waiting to be solved.

Enter **Diffblue Cover**.

Cover helps developers by using code to write code. More specifically, its AI technology uses existing Java bytecode to automatically write and maintain a set of associated unit regression tests, so that developers don't have to do all the work.

Cover can write the same amount of unit tests in **8 hours** as a developer could in **a full year**. And the output is not only complete, it's human readable. Like this, which Cover wrote in a couple of seconds:



```
@Test
void testProcessCreationForm() throws Exception {
  // Arrange
  doNothing().when(this.ownerRepository).save((Owner) any());
  MockHttpServletRequestBuilder requestBuilder = MockMvcRequestBuilders.post( urlTemplate: "/owners/new")
          .param( name: "address",  ...values: "42 Main St")
          .param( name: "city",  ...values: "Oxford")
          .param( name: "telephone",  ...values: "4105551212")
          .param( name: "firstName",  ...values: "Jane")
          .param( name: "lastName",  ...values: "Doe");

  // Act and Assert
  MockMvcBuilders.standaloneSetup(this.ownerController) StandaloneMockMvcBuilder
          .build() MockMvc
          .perform(requestBuilder) ResultActions
          .andExpect(MockMvcResultMatchers.status().isFound())
          .andExpect(MockMvcResultMatchers.model().size(1))
          .andExpect(MockMvcResultMatchers.model().attributeExists( ...names: "owner"))
          .andExpect(MockMvcResultMatchers.view().name( expectedViewName: "redirect:/owners/null"))
          .andExpect(MockMvcResultMatchers.redirectedUrl( expectedUrl: "/owners/null"));
}
```

Diffblue Cover works in one of 3 ways:

- As an **IntelliJ IDEA IDE plug-in** for specific methods, classes or packages (try the **Cover Community Edition here**);
- Via a **faster, more powerful command line interface (CLI)** that enables automatic unit test writing up to module level (use the **free trial of Developer Edition** to see what Cover can really do);
- **Scripted and added to an CI pipeline** for maximum automation and consistency at scale (it's best to **talk to someone from Diffblue** about this).

Whether you're managing legacy Java code, moving applications to the cloud, shifting left with DevOps or just want to deliver software faster, Cover can help you get the unit test coverage you need to make changes with confidence.

To find out more take a look at **www.diffblue.com**.

Joyful testing,
Philip

Copyright © 2023 rieckpil, All rights reserved.
You are receiving this email because you opted in at our website https://rieckpil.de

Our mailing address is:
rieckpil
Fließstraße 17
Berlin 12439
Germany

Add us to your address book

Want to change how you receive these emails?
You can update your preferences or unsubscribe from this list.