测试与重构

张博超

- 方润 - 程序员（Delphi/ASP.NET）
- Microsoft - SDET （C#）
- Irdeto - Lead Engineer / ScrumMaster（C#/javascript）
- 百度 - 敏捷教练
- Odd-e - 教练/顾问/培训师/程序员

咨询/教练辅导涵盖产品研发各个方面：组织/流程/团队/工程能力等

部分客户：

- GE医疗Surgery - 大规模敏捷LeSS导入 (C++/Java)
- HSBC移动 - 工程实践教练辅导 (Java/Node.js/Javascript)
- 银科控股 - 敏捷教练辅导与导入 (Node.js)
- BMW - 大规模敏捷LeSS 导入 (C++/Python)
- 一加手机 - 大规模敏捷教练辅导与导入 (Java)

回顾一下什么是重构

- 👷 **发现问题，减弱问题** - 识别代码臭味，尝试重构以降低该臭味
- 📝 **小步** - 控制每次引入的变化尽可能的小并且只干一件事情
- 🛠️ **测试** - 尽可能频繁的运行测试以保证没有破坏代码原有行为
- 🛌 **日常** - 重构是日常行为，通常不应该是一个专门的任务

测试也需要重构

- 测试的测试是什么?
  - 代码
- 重构测试的目的是什么?
  - 降低变更成本
  - 提升可读性

Cucumber

Feature

```
Feature: Calculator

  Scenario: Add
    When add 1 and 1
    Then get 2

  Scenario: Subtraction
    When subtract 1 from 2
    Then get 1
```

Steps

```
@When("add {int} and {int}")
public void add(int a, int b) {
    calculator.add(a, b);
}

@Then("get {int}")
public void get(int expected) {
    assertEquals(expected, calculator.getResult());
}

@When("subtract {int} from {int}")
public void subtract(int a, int b) {
    calculator.subtract(b, a);
}
```

啰嗦的文档


```
Feature: Login

  Background:
    Given there is a user with email "zbcjackson@odd-e.com" and password "password"

  Scenario: login_success
    When login with email "zbcjackson@odd-e.com" and password "password"
    Then login success

  Scenario: fail_incorrect_passwd
    When login with email "zbcjackson@odd-e.com" and password "incorrect-password"
    Then login fail with message "Email and password are invalid."

  Scenario: fail_nonexisting_email
    When login with email "zbcjackson@odd-e-dummy.com" and password "incorrect-password"
    Then login fail with message "Email and password are invalid."

  Scenario: fail_no_passwd
    When login with email "zbcjackson@odd-e-dummy.com"
    Then login fail with message "Password should not be empty"

  Scenario: fail_no_email
    When login with password "incorrect-password"
    Then login fail with message "Email should not be empty"
```

这样就好多了


```
Feature: Login

  Background:
    Given there is a user with email "zbcjackson@odd-e.com" and password "password"

  Scenario: login_success
    When login with email "zbcjackson@odd-e.com" and password "password"
    Then login success

  Scenario Outline: Validation
    When login with email "<email>" and password "<password>"
    Then login fail with message "<message>"
    Examples:
      | email                      | password           | message                          | case           |
      | zbcjackson@odd-e.com       | incorrect-password | Email and password are invalid. | Wrong password |
      | zbcjackson@odd-e-dummy.com | incorrect-password | Email and password are invalid. | Wrong email    |
      | zbcjackson@odd-e-dummy.com |                    | Password should not be empty    | Empty password |
      |                            | incorrect-password | Email should not be empty       | Empty email    |
```

意外的细节


```
Feature: Account

  Background:
    Given there is a user with email "zbcjackson@odd-e.com" and password "password"

  Scenario: account_add_success
    When login with email "zbcjackson@odd-e.com" and password "password"
    Then login success
    When add account "wangbb" and balance "10"
    Then account "wangbb" and balance "10" add succeed

  Scenario: account_add_fail_balance_0
    When login with email "zbcjackson@odd-e.com" and password "password"
    Then login success
    When add account "wangbb" and balance "0"
    Then account "wangbb" and balance "0" failed with empty balance

  Scenario: account_add_fail_no_name_10
    When login with email "zbcjackson@odd-e.com" and password "password"
    Then login success
    When add balance "10"
    Then failed with "Name should not be empty"

  ...
```

用钩子来隐藏细节


```
@login
Feature: Account

  Scenario: account_add_success
    When add account "wangbb" and balance "10"
    Then account "wangbb" and balance "10" add succeed

  Scenario: account_add_fail_balance_0
    When add account "wangbb" and balance "0"
    Then account "wangbb" and balance "0" failed with empty balance

  Scenario: account_add_fail_no_name_10
    When add balance "10"
    Then failed with "Name should not be empty"

  ...
```

另一种细节：第三方类库的使用

- 重复的使用细节
- 自己的接口
- 封装依赖
- 屏蔽API变更或者工具变更

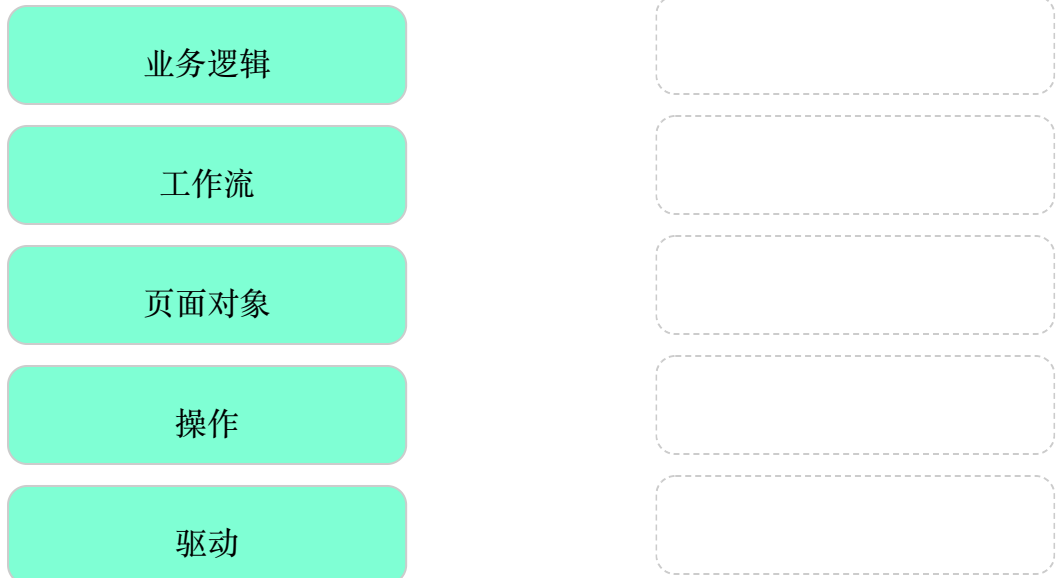
操作细节：用户交互

- 保留意图，屏蔽动作
- 屏蔽控件变化

页面样式

- 屏蔽页面变化

架构分层



单元测试

可读性第一

- case名字
- AAA
- 屏蔽其他细节

消除重复：一切恶魔的根源

鸡生蛋，蛋生鸡的问题

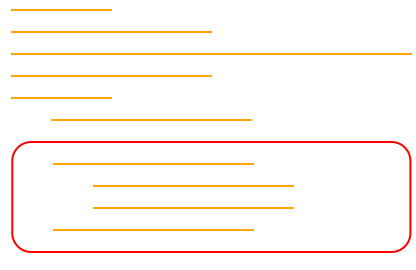
你依然还是依赖其它测试的

控制无测试的变更大小

- 利用一些设计技巧
- 利用重构工具

创造接缝(Seam)

- 处理依赖
  - 隔离依赖
  - 注入依赖
- 处理多职责混杂长方法
  - 提取相对独立逻辑到独立类
  - 测试提取的独立类
  - 探寻独立业务概念



```
public class Seam {


    public void execute() {
        
    }
}
```

谢谢

wechat/twitter/gmail : zbcjackson