



AUTOMATED ACCEPTANCE TESTING

WITH CODECEPTION

PHP Dorset / October 2016

IAIN POULSON

[@polevaultweb](#)

IAIN POULSON

@polevaultweb



<https://www.meetup.com/Bournemouth-WordPress-Group/>

27th October

[@wpbournemouth](#)



Delicious Brains



WP Migrate DB Pro





WP Migrate DB Pro



WP Offload S3





WP Migrate DB Pro



WP Offload S3



mergebot

THE PROBLEM

Releasing a new version of one of these plugins takes time

WP OFFLOAD S3

Manual Acceptance Test Scripts

Lite free plugin	77
Premium plugin	81
AWS plugin dependency	20
6 addon plugins	181

3 x Developer rounds of testing with bug fixes and retests

WHAT'S THE SOLUTION?

WHAT'S THE SOLUTION?

Hire a tester or testers for the team

WHAT'S THE SOLUTION?

~~Hire a tester or testers for the team~~

WHAT'S THE SOLUTION?

~~Hire a tester or testers for the team~~

Hire the services of an external Quality Assurance team

WHAT'S THE SOLUTION?

~~Hire a tester or testers for the team~~

~~Hire the services of an external Quality Assurance team~~

WHAT'S THE SOLUTION?

~~Hire a tester or testers for the team~~

~~Hire the services of an external Quality Assurance team~~

Hire more developers

WHAT'S THE SOLUTION?

~~Hire a tester or testers for the team~~

~~Hire the services of an external Quality Assurance team~~

~~Hire more developers~~

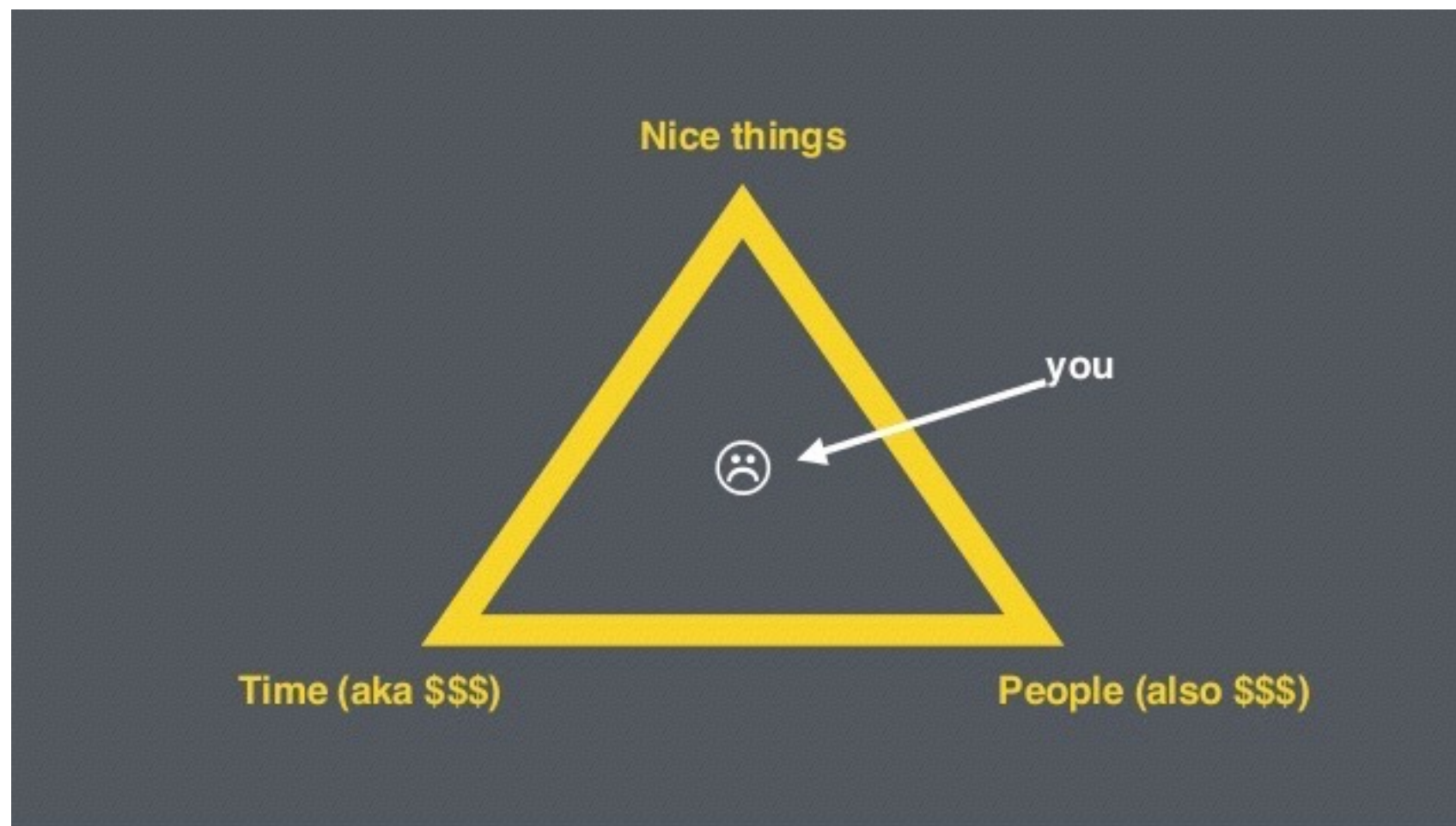
THE EUREKA MOMENT

THE EUREKA MOMENT

Rock Solid Software Testing Without Hiring an Army

Trish Khoo

WHY WE CAN'T HAVE NICE THINGS



DO NOT HIRE A QA TEAM

The problem with this is that now you've got a team of people who is accountable for quality in your software and they have no power to affect the quality of it directly

HER SOLUTION

AUTOMATION

HER SOLUTION

AUTOMATION

■ The problem is that usually this is approached in a way that isn't terribly scalable

HER SOLUTION

IN-HOUSE AUTOMATION

The whole team has to commit to quality

**WHAT DOES THAT ACTUALLY
MEAN?**

THE PLAN

1. One developer works on replacing manual with automated tests
2. All new code should have corresponding tests written
3. Code changes should have tests updated
4. PRs should have suite of all tests run before review

THE RESULT

1. Regression issues are caught much earlier
2. Release testing becomes much shorter, less labour intensive
3. Manual tests can be reduced to a skim test
4. Releases can happen more often

AUTOMATE ALL THE THINGS



CODECEPTION

Behaviour-driven testing framework

Built on top of PHPUnit

Runs acceptance, functional, and unit tests

ACCEPTANCE TESTS

Acceptance tests can cover standard but complex scenarios from a user's perspective. With acceptance tests you can be confident that users, following all defined scenarios, won't get errors.

<http://codeception.com/docs/01-Introduction>

A SAMPLE TEST

```
<?php
$I = new AcceptanceTester($scenario);
$I->amOnPage('/');
$I->click('Sign Up');
$I->submitForm('#signup', ['username' => 'MilesDavis', 'email' => 'miles@davis.com']);
$I->see('Thank you for Signing Up!');
```

MODULES

Database

Filesystem

PhpBrowser

Lots more

PHPBROWSER

A PHP web scraper, which acts like a browser: it sends a request, then receives and parses the response

WEBDRIVER

Selenium

PhantomJS

WORDPRESS INTEGRATION

WPBrowser

Extends PhpBrowser and DB Module with WordPress specific assertions

```
1 <?php
2 $I = new AcceptanceTester( $scenario );
3
4 // Login to wp-admin
5 $I->loginAsAdmin();
6
7 // Navigate to the Media Library
8 $I->amOnPage( '/wp-admin/media-new.php' );
9 $I->waitForText( 'Upload New Media' );
10
11 // 📁 Add new file
12 $I->attachFile( 'input[type="file"]', 'team.jpg' );
13
14 // Wait for upload
15 $I->waitForElement( '.edit-attachment', 20 );
16 $I->seeElement( '.edit-attachment' );
17 $I->click( '.edit-attachment' );
18
19 // Navigate to the Edit Media window
20 $I->executeInSelenium( function ( \Facebook\WebDriver\Remote\RemoteWebDriver $webdriver ) {
21     $handles = $webdriver->getWindowHandles();
22     $last_window = end( $handles );
23     $webdriver->switchTo()->window( $last_window );
24 } );
25 $I->waitForText( 'Edit Media' );
26
27 // Check URL is an S3 one
28 $url = $I->grabValueFrom( 'attachment_url' );
29 $this->assertContains( 'amazonaws.com', $url );
30
31 // Parse the URL
32 $url = explode( '.com/', $url );
33 $url_parts = explode( '/', $url[1] );
34
35 $bucket = array_shift( $url_parts );
36 $key = implode( '/', $url_parts );
37
38 // Check attachment has been offloaded to Amazon S3
39 $I->setBucket( $bucket )->seeFile( $key );
40
41
```

MacBook-Pro:~

Project: wpos3-acceptance-testing-example

run-tests.sh

- wpos3-acceptance-testing-example /Applications/MAMP/htdocs/wpos3-
 - tests
 - vendor
 - .gitignore
 - codeception.dist.yml
 - codeception.yml
 - composer.json
 - composer.lock
 - README.md
 - run-tests.sh

```
1 #!/usr/bin/env bash
2
3  function usage() {
4      echo "Usage: $0 -u dbuser -p dbpassword [ -h dbhost ] [ -s start from scratch ] [ -t path to"
5      exit 2
6  }
7
8  function parse_yaml {
9      local prefix=$2
10     local s1=${1//%[:%]} s2=${1//%[:%]} s3=${1//%[:%]} s4=${1//%[:%]} s5=${1//%[:%]} s6=${1//%[:%]} s7=${1//%[:%]} s8=${1//%[:%]} s9=${1//%[:%]} s10=${1//%[:%]} s11=${1//%[:%]} s12=${1//%[:%]} s13=${1//%[:%]} s14=${1//%[:%]} s15=${1//%[:%]} s16=${1//%[:%]} s17=${1//%[:%]} s18=${1//%[:%]} s19=${1//%[:%]} s20=${1//%[:%]} s21=${1//%[:%]} s22=${1//%[:%]} s23=${1//%[:%]} s24=${1//%[:%]} s25=${1//%[:%]} s26=${1//%[:%]} s27=${1//%[:%]} s28=${1//%[:%]} s29=${1//%[:%]} s30=${1//%[:%]} s31=${1//%[:%]} s32=${1//%[:%]} s33=${1//%[:%]} s34=${1//%[:%]} s35=${1//%[:%]} s36=${1//%[:%]} s37=${1//%[:%]} s38=${1//%[:%]} s39=${1//%[:%]} s40=${1//%[:%]} s41=${1//%[:%]} s42=${1//%[:%]} s43=${1//%[:%]} s44=${1//%[:%]} s45=${1//%[:%]} s46=${1//%[:%]} s47=${1//%[:%]} s48=${1//%[:%]} s49=${1//%[:%]} s50=${1//%[:%]} s51=${1//%[:%]} s52=${1//%[:%]} s53=${1//%[:%]} s54=${1//%[:%]} s55=${1//%[:%]} s56=${1//%[:%]} s57=${1//%[:%]} s58=${1//%[:%]} s59=${1//%[:%]} s60=${1//%[:%]} s61=${1//%[:%]} s62=${1//%[:%]} s63=${1//%[:%]} s64=${1//%[:%]} s65=${1//%[:%]} s66=${1//%[:%]} s67=${1//%[:%]} s68=${1//%[:%]} s69=${1//%[:%]} s70=${1//%[:%]} s71=${1//%[:%]} s72=${1//%[:%]} s73=${1//%[:%]} s74=${1//%[:%]} s75=${1//%[:%]} s76=${1//%[:%]} s77=${1//%[:%]} s78=${1//%[:%]} s79=${1//%[:%]} s80=${1//%[:%]} s81=${1//%[:%]} s82=${1//%[:%]} s83=${1//%[:%]} s84=${1//%[:%]} s85=${1//%[:%]} s86=${1//%[:%]} s87=${1//%[:%]} s88=${1//%[:%]} s89=${1//%[:%]} s90=${1//%[:%]} s91=${1//%[:%]} s92=${1//%[:%]} s93=${1//%[:%]} s94=${1//%[:%]} s95=${1//%[:%]} s96=${1//%[:%]} s97=${1//%[:%]} s98=${1//%[:%]} s99=${1//%[:%]} s100=${1//%[:%]}

```

Terminal

MacBook-Pro:wpos3-acceptance-testing-example igp\$ sh run-tests.sh

Resetting WordPress Database...

6: TODO 9: Version Control Terminal Event Log

TIP OF THE ICEBERG

Lots of different modules

Continuous Integration

BDD support

DOWNSIDERS

1. Large overhead in environment setup and converting manual tests
2. Potential for customisation of modules to fit our specific needs
3. Code changes, so the tests will need to be maintained

RESOURCES

<http://businessofsoftware.org/2016/06/rock-solid-software-testing-without-hiring-an-army-trish-khoo-google-bos2015/>

<http://codeception.com/>

<https://github.com/lucatumel/wp-browser>

<https://github.com/polevaultweb/wpos3-acceptance-testing-example>

<https://deliciousbrains.com/codeception-automate-wordpress-plugin-testing/>

ANY QUESTIONS?