



# Make Your Code Behave!

Jason Endres

# Sponsors



# About Me

- @tinyjason
- B.S from Herzing University
- ISP, Network Admin, USGS, Startup (UI Dev)
- 6 years at Trek
- League of Legends, Warhammer 40k

Trek

We are hiring!



# What we will cover

- BDD (Behavior-driven development)
- Gherkin
- Behat
  - Used with phantomjs
- Code Examples

# Behavior Driven Development

- Collaboration between QA, Dev's, and the Business
- Evolved from TDD (Test Driven Development)
  - Non programmer friendly
- Works best in an agile environment
  - Focus on deliverables
  - Limits the number of scenarios per task/story



# Cont.

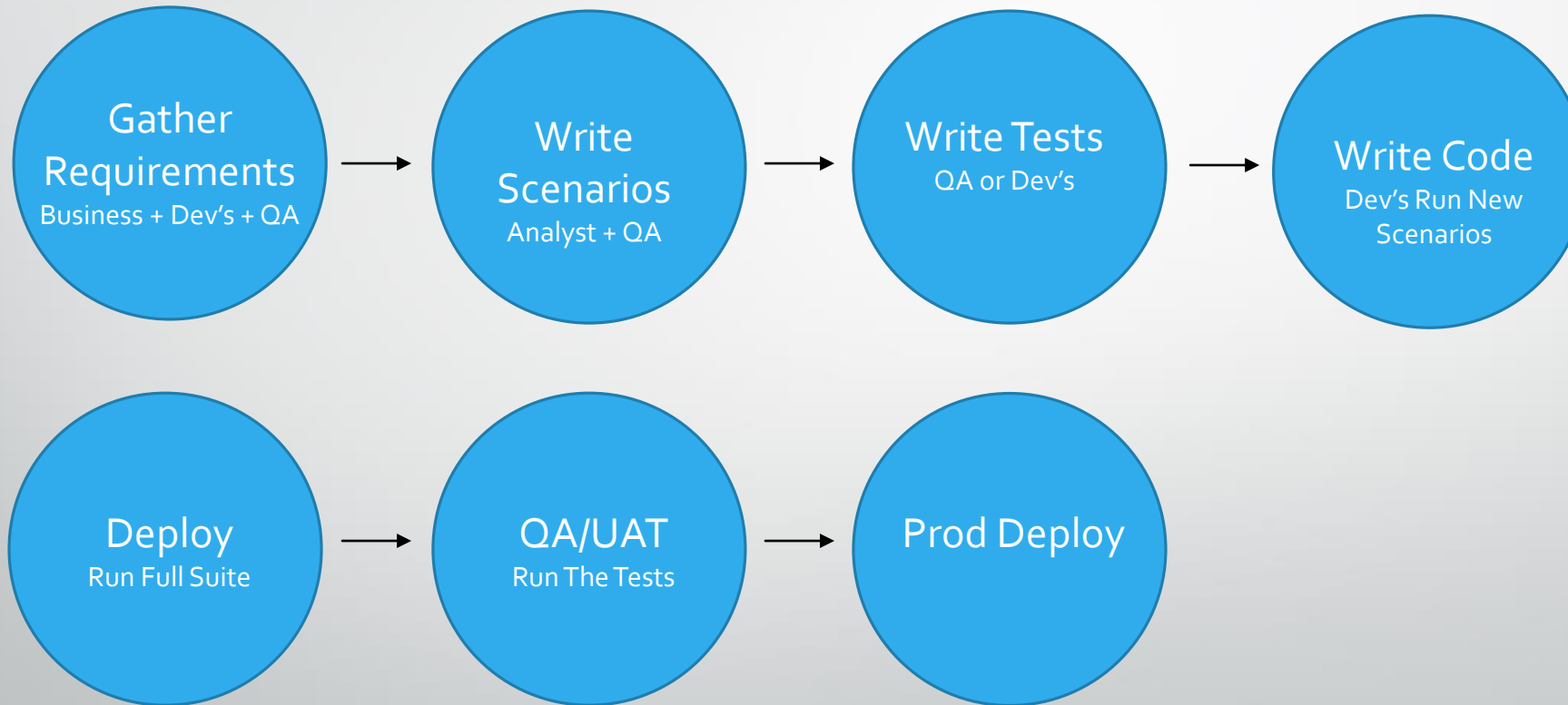
- Well established goals, requirements documentation
- Involve the stakeholders (outside in software development)
- Automation
- Regression testing

# Business Value / Deliverables

- Work on small pieces
  - Meet the requirements 1 piece at a time
  - Avoid high cost scope creep
- Each thing should be releasable
  - Feature toggling
- Minimize re-work



# Process



# Value Even if not Ideal

- Can run tests against your site outside of your codebase
  - Take advantage use Composer
- Write scenario's and tests yourself
  - The process can be agile even if your team is not
  - Ignore the "Has to be" mentality
- Start small
  - Basic go here is there content
- Don't have to be agile

# Use a CI solution


- Jenkins, etc
- Automated deploys
- Execute your test suite
- Easy verification without developer work

# Gherkin

- Business Readable
- Documentation
- Automated Testing
- Identify what your code should do

# Feature

- .feature file
- Organize feature files
- Contains 1 or many scenarios
- List acceptance criteria
- Can represent a deliverable



Feature: Some terse yet descriptive text of what is desired  
Textual description of the business value of this feature  
Business rules that govern the scope of the feature  
Any additional information that will make the feature easier  
to understand

# Scenarios

- A list of steps to validate the scenario
- Usually multiple scenario's per feature
- Business situation



# Examples

Scenario: Some determinable business situation

Scenario: Submit support request

Scenario: Buy a product





# Steps

- These define how you achieve the scenario
- Make sure they make sense
- These can help identify need for more scenarios

# Example

Given some precondition  
And some other precondition  
When some action by the actor  
And some other action  
And yet another action  
Then some testable outcome is achieved  
And something else we can check happens too



# Given, When, Then, But or And.

There keywords are treated the same.  
Just because you can doesn't mean you should.

# Successful steps

- Has to find a matching step definition
- Executing the definition does not throw an exception

# Undefined steps

- The step definition is not found
- All subsequent steps are skipped
- You will get output of an example definition to create

# Failed steps

- Definition was found but execution yielded an exception
- No other steps are executed

# Behat

- PHP BDD framework (5.3, 5.4)
- Executable ran from command line

# Installation - Composer

```
{  
    "require": {  
        "behat/behat": "2.4.*@stable"  
    },  
    "minimum-stability":  
        "dev",  
    "config": {  
        "bin-dir": "bin/"  
    }  
}
```



# Required PHP Modules

- Mbstring
- xml

# PhantomJs

- Headless web browser
- `npm install phantomjs`
- Start phantomjs: `phantomjs --webdriver=8643`

# More Packages Needed For Phantom

```
{
  "require": {
    "behat/behat": "2.4.*@stable",
    "behat/mink": "1.4.*@stable",
    "behat/mink-extension": "*",
    "behat/mink-goutte-driver": "*",
    "behat/mink-selenium2-driver": "*"
  },
  "minimum-stability": "dev",
  "config": {
    "bin-dir": "bin/"
  }
}
```

# Behat.yaml

phantomjs:

context:

class: 'FeatureContext'

extensions:

Behat\MinkExtension\Extension:

goutte: ~

selenium2:

wd\_host: "http://localhost:8643/wd/hub"

# FeatureContext

- User defined functions
- Can be re-used
- 1 function called for each step

# Make Feature Context Use Mink

- `use Behat\MinkExtension\Context\MinkContext;`
- `class FeatureContext extends MinkContext`

# Executing Your Tests

- `bin/behata -p phantomjs`
- This will find the `behata.yml` file in your directory
- It will execute all feature files

# Xpath

```
public function iClickOnTheElementWithXPath($xpath) {  
    $session = $this->getSession();  
    // get the mink session  
    $element = $session->getPage()->find( 'xpath', $session->getSelectorsHandler()->selectorToXPath('xpath', $xpath) );  
    // runs the actual query and returns the element  
    // errors must not pass silently  
    if (null === $element) {  
        throw new \InvalidArgumentException(sprintf('Could not evaluate XPath: "%s"', $xpath));  
    }  
    // ok, let's click on it  
    $element->click();  
}
```



# CSS Selector

```
public function iClickOnTheElementWithCSSSelector($cssSelector) {  
    $session = $this->getSession();  
    $element = $session->getPage()->find( 'xpath', $session->getSelectorsHandler()->selectorToXpath('css', $cssSelector));  
    if (null === $element) {  
        throw new \InvalidArgumentException(sprintf('Could not evaluate CSS Selector: "%s"', $cssSelector));  
    }  
    $element->click();  
}
```

# Execute Javascript

```
$session = $this->getSession();  
$script = '$("#search_form").submit();';  
$session->executeScript( $script );
```

# Responsive Testing

Set the phantom window size with a step in the scenario

```
public function resizeWindow($width, $height) {  
    $this->getSession()->resizeWindow($width, $height, 'current');  
}
```

# Tips

- You can specify the config file with the --config
  - Allows you to create multiple files with useful names
  - Can allow you to execute your tests with different base urls
  - Allows you to specify different BehatContext files

## Cont.

- Specify which feature files to execute
  - Run behat passing in a folder or a specific feature file

# Debugging

- Phantomjs server window
- `getCurrentUrl()`
- `getPage()`
- Verify where you are



Questions?

# Rate My Talk Please

- <https://joind.in/talk/view/11739>



# Resources

- <https://pythonhosted.org/pytest/philosophy.html>
- <https://github.com/cucumber/cucumber/wiki/Gherkin>
- <https://github.com/Behat/MinkExtension-example/tree/master/features>
- <http://docs.behat.org/en/v2.5/>
- <http://blog.lepine.pro/images/2012-03-behat-cheat-sheet-en.pdf>