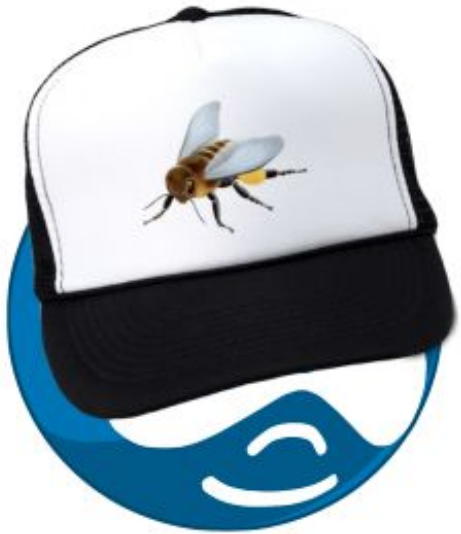
SRIJAN

Testing with Behat (Php based framework)

Deepshikha Singh
Soumyajit Basu

Objectives

- Background
- Run Behat
- Write Script
- Expand Behat
- Best Uses



SRIJAN

The Dream

SRIJAN

Who should use Behat?

SRIJAN

- Behat was developed for Business Analysts (BA) and developers, so as to bridge the gap between business stakeholders and the development team

- It is also used by testers for testing websites. As Behat uses Gherkin language to write it’s scenarios, it becomes easy for testers to map it with the acceptance criteria of the project.

When to use Behat?

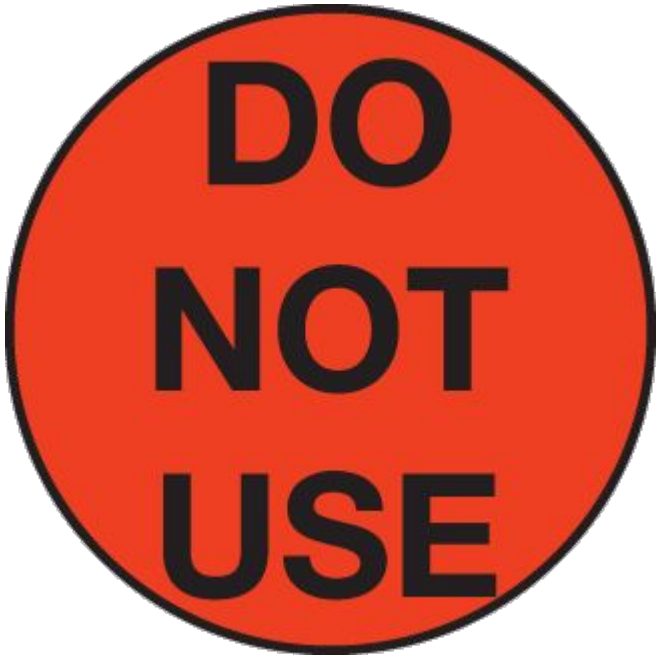
SRIJAN

- Testing the data/content on the webpage
- Testing actions (like links, buttons etc.)
- Testing forms
- In migration state of any website from one CMS to other
- For end to end testing (i.e. flow of an application from start to finish)
- For Functional, Regression, Smoke and Sanity Testing
- For running the test cases quickly using Goutte driver

When not to use Behat?

SRIJAN

- Dynamic data
- Images
- Frequent changes in website
- Filling up forms in production site
- Http response of links throughout the website



Behat is…

- Is an open source behavior-driven development framework
- Allows you to write “human-readable” tests
- Integrates perfectly with Drupal

https://www.drupal.org/project/drupalextension

Interesting!

Behat is based off of Ruby’s Cucumber

SRIJAN

“Human-Readable”?

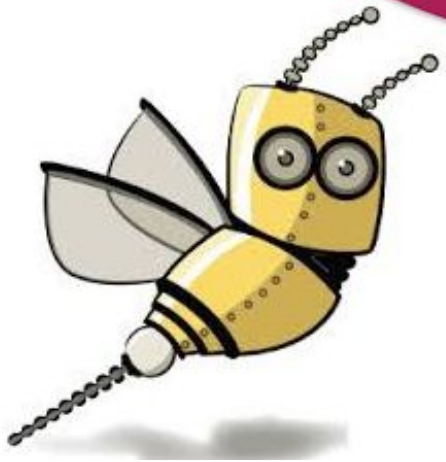
- Describe exactly what you are testing in your own language
- Gherkin language
- Provides a usable and searchable list of scenarios/scenario structure
- Always start with following scenario

keywords(given, when, then, and, but)

Is your language Supported?

$ behat --story-syntax --lang YOUR_LANG

Don’t forget to include your language in you’re my-feature.feature



SRIJAN

Cause and Effect

A test is made up of a

Feature – describe your goals for the test

Scenario – layout the the path the test will follow

Scenario outline – test multiple variables in the same test

```
Scenario: Some determinable business situation
    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
```

Steps

SRIJAN

Every scenario we lay out will start with key words. These are the steps of our feature

- Given – known state
- When – key action
- Then - observe
- And , But – expand on state, action and observation

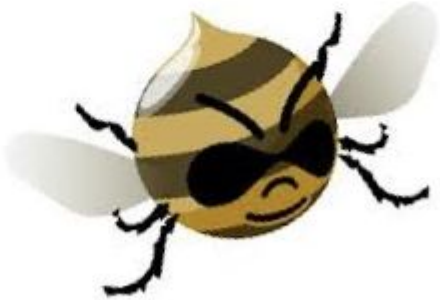
```
Scenario: Multiple Givens
  Given one thing
    And an other thing
    And yet an other thing
  When I open my eyes
  Then I see something
  But I don't see something else
```

Statements in Code

SRIJAN

Given

```
/**
 * Moves user to the specified path.
 *
 * @Given /^I am in the "([^"]*)" path$/
 *
 * @param   string $path
 */
public function iAmInThePath($path)
{
    $this->moveToNewPath($path);
}
```



Expand Functionality

SRIJAN

Add your own statements in the FeatureContext.php

```
/**
     * @Given /^I set browser window size to "([^"]*)" x
"([^"]*)"$/
     */

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

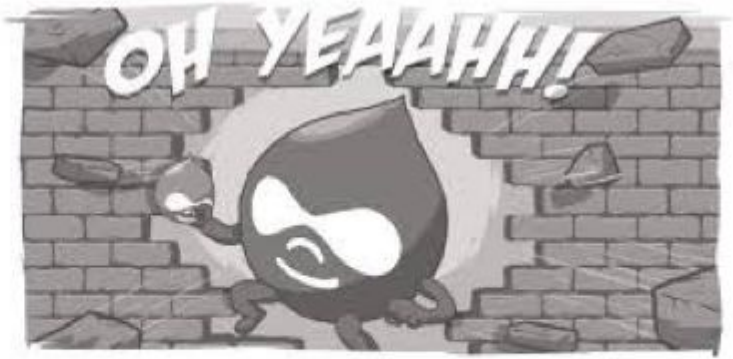
```
@browser
 Scenario Outline: Verify responsive menu
   Given I set browser window size to <width> x
<height>
   Given I am on "/"
   Then I should see "Menu"
   And I click ".responsive-menus responsive-
menus-0-0"
   Then I should see <content> in the <selector>
element


   Examples:
     | content    | selector           | width | height
|
     | "Menu"     | "span.toggler"     | "615" | "800"
|
     | "Menu"     | "span.toggler"     | "480" | "800"
|
     | "Search"   | ".search-block-form" | "615" |
"800"  |
     | "Search"   | ".search-block-form" | "480" |
"800"  |
```

Useful you ask?

SRIJAN

- Behat integrates with Selenium and other browser emulators
- Easy to pick up
- Can be added to any project any time
- Can fit any project
- Can ease post launch task work load once implemented

Don't forget to start selenium
./start_selenium.sh



Mink

SRIJAN

Goes hand and hand with Behat and Drupal

Mink removes API differences between different browser emulators

Web apps (Mink)	
Surf:	Assertions:
I am on "url"	I should see "content"
I go to "url"	the response should contain "content"
I reload the page	I should not see "content"
I move backward one page	the response should not contain "content"
I move forward one page	the "form_element" field should contain "value"
I press "button"	the "form_element" field should not contain "value"
I follow "link"	the "form_checkbox" checkbox should be checked
	the "form_checkbox" checkbox should not be checked
Forms:	
I fill in "form_element" with "value"	I should be on "page"
I fill in "value" for "form_element"	the url should match "url"
I fill in the following	the "num_position" element should contain "value"
I select "form_option" from "form_select"	I should see "value" in the "element" element
I additionally select "form_option" from "form_select"	I should see an "element" element
I check "form_checkbox"	I should not see an "element" element
I uncheck "form_checkbox"	I should see number "element" elements

Resources

Quick Intro to Behat

http://docs.behat.org/en/v2.5/quick_intro.html

Gherkin Syntax

http://docs.behat.org/en/v2.5/guides/1.gherkin.html

Nice Tutorial

http://code.tutsplus.com/tutorials/bdd-with-behat--net-36171

Step Definitions

http://docs.behat.org/en/v2.5/guides/2.definitions.html

Testing Features Guide

http://docs.behat.org/en/v2.5/guides/4.context.html

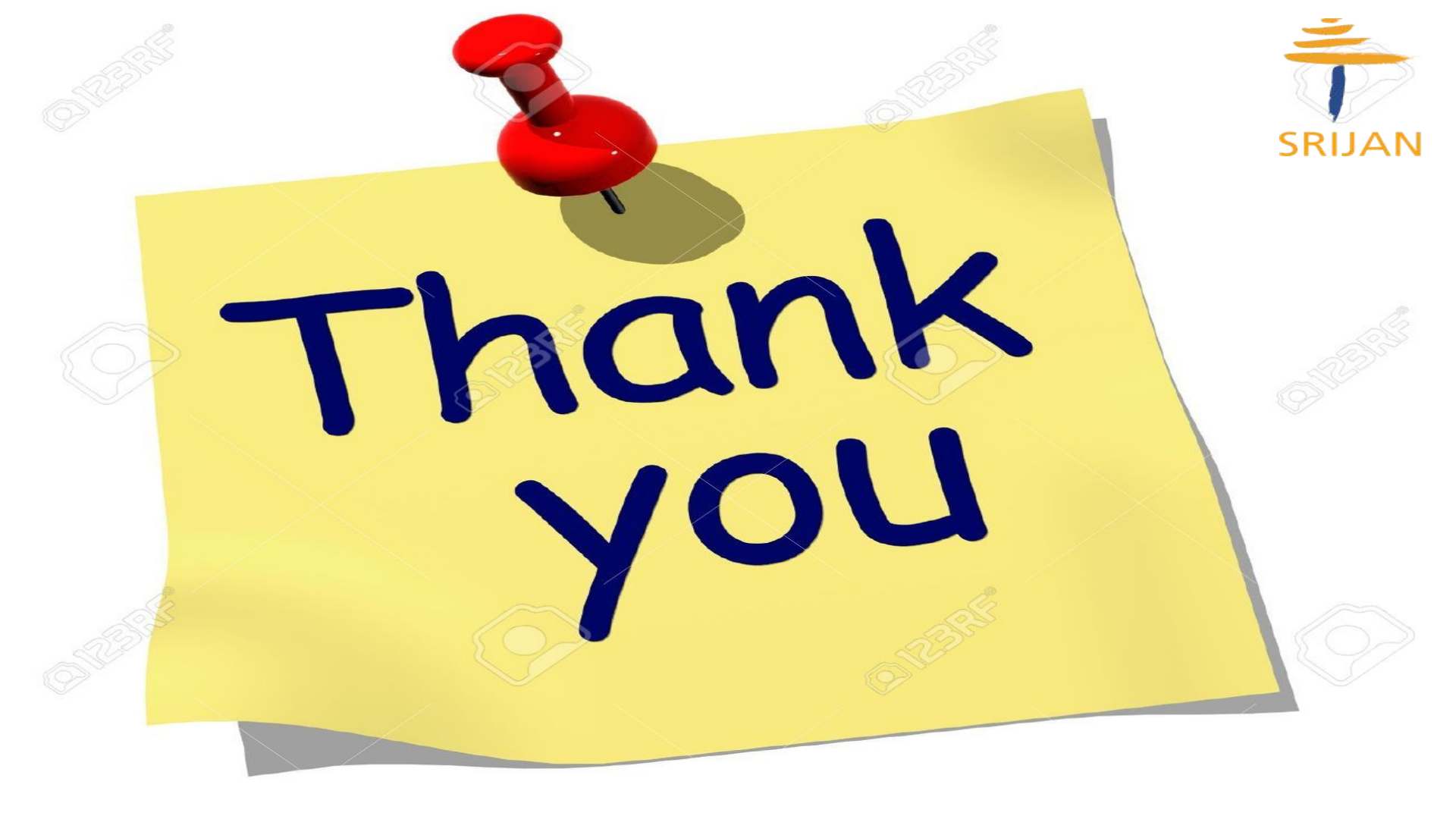
Behat Cheatsheet

http://blog.lepine.pro/images/2012-03-behat-cheat-sheet-en.pdf

SRIJAN

Questions

SRIJAN


Thank
you