Testing JavaScript with Jasmine

Carolyn Cole
cam156@psu.edu

Hector Correa
hjc14@psu.edu

The Pennsylvania State University

The Problem

Testing JavaScript with Ruby tools is hard

- Feature Tests
  - tend to be slow
  - hard to test error cases
- View Tests
  - do not test JavaScript

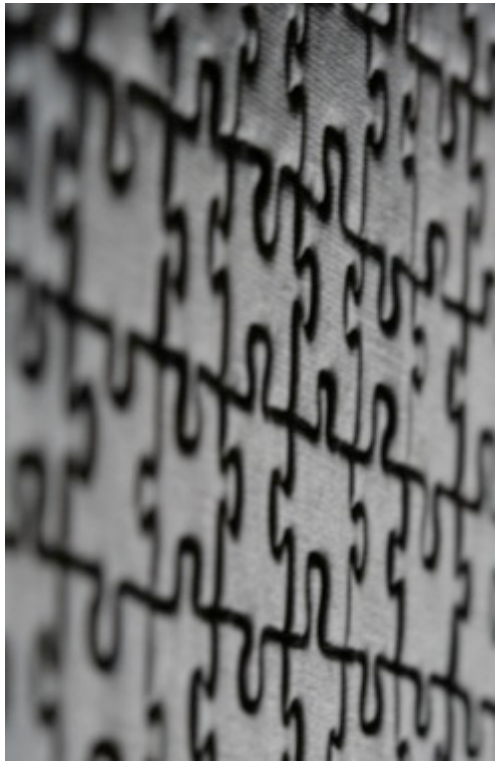
Quick Demo

https://github.com/hectorcorrea/riceonrails

The Solution

- Integrate a JavaScript Testing Framework

- There are many options
  - Karma, QUnit, Buster, Mocha.js, **Jasmine**, …
  - http://stackoverflow.com/questions/300855/javascript-unit-test-tools-for-tdd



Why Jasmine?

- Easily integrated with existing tests
- Familiar syntax to RSpec
- Built in matchers
- Mock objects
- Control time

Familiar Syntax - JavaScript

Ruby/Rspec

```
describe "something" do

  before do
    ...
  end

  it "does something" do
    ...
  end

end
```

JavaScript/Jasmine

```
describe("something", function() {

  beforeEach(function() {
    ...
  });

  it("does something", function() {
    ...
  });

});
```

Familiar Syntax - CoffeeScript

Ruby/Rspec

```
describe "something" do

  before do
    ...
  end

  it "does something" do
    ...
  end

end
```

CoffeeScript/Jasmine

```
describe "something", ->

  beforeEach ->
    ...

  it "does something", ->
    ...
```

Running Jasmine

- User Interface

  ```
  $ rake jasmine
  ```

  > go to localhost:8888 to view results

- Continuous Integration

  ```
  $ rake jasmine:ci
  ```

  ```
  Waiting for jasmine server on 58458...
  jasmine server started
  ..
  2 specs, 0 failures
  [2015-09-21 10:15:15] INFO  going to shutdown ...
  [2015-09-21 10:15:15] INFO  WEBrick::HTTPServer#start done
  ```

Testing plain vanilla JavaScript

```
// Calculator.js
Calculator.addNumbers = function(a, b) {
  return a + b;
}

// CalculatorSpec.js
describe("Calculator", function() {
  it("adds two numbers", function() {
    result = Calculator.addNumbers(4,5);
    expect(result).toEqual(9);
  });
});
```

Testing DOM elements (1/2)

```
// Calculator.js
Calculator.updateUI = function() {
  var a = $("#number1").val();
  var b = $("#number2").val();
  var result = Calculator.addNumbers(a, b);
  $("#resultGoesHere").text(result);
}
```

Testing DOM elements (2/2)

```
// CalculatorUISpec.js
describe("Calculator UI", function() {
  it("updates the UI with the result", function() {
    var html = '<input id="number1" value="4"/> ' +
           '<input id="number2" value="3"/> ' +
           '<span id="resultGoesHere"/>';
    setFixtures(html);

    Calculator.updateUI();

    var result = $("#resultGoesHere");
    expect(result.text()).toEqual("7");
  });
});
```

Expect and matchers

- Syntax similar to Rspec

```
expect(result.text()).toEqual("7");
```

```
var button = $("#theButton");
expect(button).toBeDisabled();
```

Matchers

- Built-in
  - `toMatch, not.toMatch, toBeDefined, not.toBeDefined, toBeNull, toBeTruthy, toContain, toEqual, ...`
  - http://jasmine.github.io/2.0/introduction.html
- Third party matchers (e.g. Jasmine jQuery)
  - `$("#someID")`
  - `toBeDisabled, not.toBeDisabled, ...`
  - https://github.com/velesin/jasmine-jquery
- Write your own

Mock Objects - Spies

- Spies can be used for any JavaScript function
  - All calls and arguments to a spy are tracked
  - It only exists in the block and will be removed after each test
  - http://jasmine.github.io/2.0/introduction.html#section-Spies

See autocomplete example in Sufia
https://github.com/projecthydra/sufia/blob/master/spec/javascripts/autocomplete_spec.js.coffee

Control Time

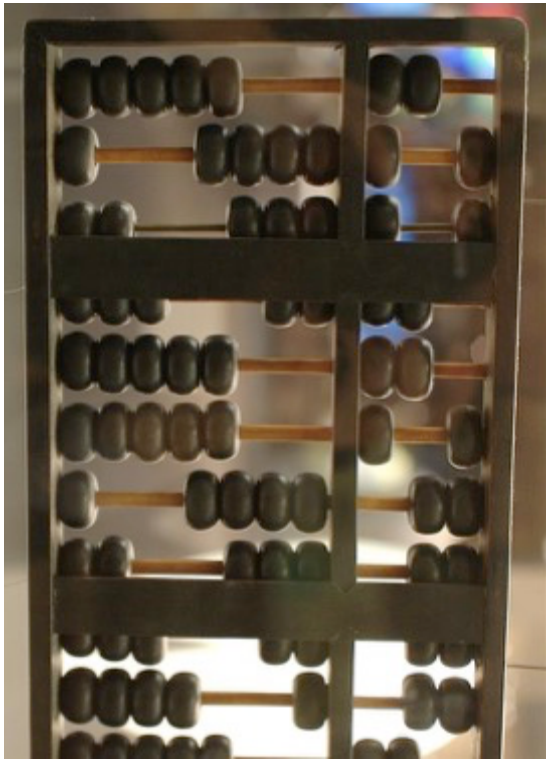
- `jasmine.clock().install`
  - Starts and defines the Jasmine clock
  - Makes calls to timer functions synchronous

- `jasmine.clock().tick`
  - Allows you to move time forward instantaneously

See autocomplete example in Sufia
https://github.com/projecthydra/sufia/blob/master/spec/javascripts/autocomplete_spec.js.coffee

Jasmine add on to RSpec

- Run Jasmine in CI mode
- Capture the output
- Evaluate the output for failures
- Existing code is available in Sufia
  - https://github.com/projecthydra/sufia/blob/master/spec/javascripts/jasmine_spec.rb
  - Would like to create a Gem or integrate with another Gem



Thanks!

Slides/code examples

https://github.com/hectorcorrea/riceonrails