



Testing Stuff With Chimp

What We'll Cover

- What is Chimp?
- Why Chimp?
- Demo

(lazy, I know)

What is Chimp

- Automated Testing tool built with node.js

```
chimp --watch
```

- Write acceptance / end-to-end tests how you like
 - Mocha
 - Jasmine
 - Cucumber.js
- Browser automation:
 - Selenium (including cloud based services)
 - WebDriverIO API

What is Chimp – Assertions

- BDD/TDD assertions:
 - Chai
 - Jasmine

Should

```
chai.should();

foo.should.be.a('string');
foo.should.equal('bar');
foo.should.have.length(3);
tea.should.have.property('flavors')
  .with.length(3);
```

Expect

```
var expect = chai.expect;

expect(foo).to.be.a('string');
expect(foo).to.equal('bar');
expect(foo).to.have.length(3);
expect(tea).to.have.property('flavors')
  .with.length(3);
```

Assert

```
var assert = chai.assert;

assert.typeOf(foo, 'string');
assert.equal(foo, 'bar');
assert.lengthOf(foo, 3);
assert.property(tea, 'flavors');
assert.lengthOf(tea.flavors, 3);
```

What is Chimp - JS

- Write it in JavaScript

```
module.exports = function() {  
  
  this.Given(/^I have visited Google$/, function () {  
    // Write the automation code here  
    pending();  
  });  
  
  this.When(/^I search for "([^"]*)"$/, function (arg1) {  
    // Write the automation code here  
    pending();  
  });  
  
  this.Then(/^I see "([^"]*)"$/, function (arg1) {  
    // Write the automation code here  
    pending();  
  });  
  
}
```

What is Chimp - Watch

- Watch mode
 - Reruns only the specs you tag
 - Focus on the feature you're working on

```
chimp --watch
```

```
@watch  
Scenario: Search for Xolv.io
```



What is Chimp - Bullet points

- Browser Session Reuse
- Downloads and Configures Dependencies
 - Selenium
 - ChromeDriver
 - IEDriver
 - PhantomJS
- Multiplatform
 - Windows
 - OSX
 - Linux

What is Chimp - Mocha

- Mocha - develop

```
chimp --mocha --watch
```

```
describe('Chimp Mocha', function() {  
  describe('Page title', function () {  
    it('should be set by the Meteor method @watch', function () {  
      browser.url('http://www.google.com');  
      expect(browser.getTitle()).to.equal('Google');  
    });  
  });  
});
```

- Mocha - test

```
chimp --mocha --browser=firefox
```


What is Chimp - Jasmine

- Jasmine - develop

```
chimp --jasmine --watch
```

```
describe('Chimp Jasmine', function() {  
  describe('Page title', function () {  
    it('should be set by the Meteor method @watch', function () {  
      browser.url('http://www.google.com');  
      expect(browser.getTitle()).toEqual('Google');  
    });  
  });  
});
```

- Jasmine - test

```
chimp --jasmine
```

What is Chimp - Cucumber

- Cucumber - develop

```
chimp --watch
```

What is Chimp - Gherkin

- Human readable feature files

```
Feature: Search the Web
```

```
  As a human
```

```
  I want to search the web
```

```
  So I can find information
```

```
  Scenario: Search for Xolv.io
```

```
    Given I have visited Google
```

```
    When I search for "Xolv.io"
```

```
    Then I see "Xolv.io"
```

What is Chimp - Helping with stuff

- Helps you write your unimplemented feature step definition files

```
1 scenario (1 undefined)
3 steps (3 undefined)
```

You can implement step definitions `for` undefined steps with these snippets:

```
this.Given(/^I have visited Google$/, function () {
  // Write the automation code here
  pending();
});
```

```
this.When(/^I search for "([^"]*)"$/, function (arg1) {
  // Write the automation code here
  pending();
});
```

What is Chimp - Step Definitions

- Here are the step definitions

```
module.exports = function() {  
  
  this.Given(/^I have visited Google$/, function () {  
    // Write the automation code here  
    pending();  
  });  
  
  this.When(/^I search for "([^"]*)"$/, function (arg1) {  
    // Write the automation code here  
    pending();  
  });  
  
  this.Then(/^I see "([^"]*)"$/, function (arg1) {  
    // Write the automation code here  
    pending();  
  });  
  
}
```

What is Chimp – WebdriverIO

- Browser interactions using WebdriverIO

```
this.Given(/^I have visited Google$/, function () {  
  browser.url('http://google.com');  
});
```

```
this.When(/^I search for "([^"]*)"$/, function (searchTerm) {  
  browser.setValue('input[name="q"]', searchTerm);  
  browser.keys(['Enter']);  
});
```

```
this.Then(/^I see "([^"]*)"$/, function (link) {  
  browser.waitForExist('a=' + link);  
});
```

What is Chimp - Meteor

- Meteor ddp connection

```
# Cucumber mode
cd tests
chimp --ddp=http://localhost:3000 --watch --path=tests

# Mocha mode
chimp --ddp=http://localhost:3000 --watch --mocha --path=tests
```

- Meteor server calls

```
// using call
var result = server.call('fixtures/resetMyApp');

// or apply
var result = server.apply('myMethodWithArgs', [argsArray]);
```

What is Chimp - Meteor

- Meteor client calls

```
var getUserProfileProperty = function (property) {  
  return Meteor.user().profile[property];  
};  
client.url('http://localhost:3000'); // the client must be on a Meteor ap  
p page  
var userName = client.execute(getUserProfileProperty, 'name').value;  
console.log(userName);
```


Why Chimp - Magic

- Cucumber + Jasmine + WebDriverIO = clean automation steps

```
this.Then(/^they see the heading "([^"]*)"$/, function(heading) {  
  expect(browser.getText('h1')).toEqual(heading);  
});
```

```
var doesNotExist = browser.waitForExist("selector", undefined, true)  
  
expect(doesNotExist).toBe(true);
```

Why Chimp - Magic

- Multi-browser

```
export CUCUMBER_BROWSERS=2
```

```
Scenario: Login into the system
```

```
  Given Both go to "/"
```

```
  Then Alice sees "Very nice" label
```

```
  And Bob sees "Very nice" label
```

```
this.Then(/^([\^ ]*) sees? "([\^"]*)" label$/, function (person, label) {  
  var _labelSelector = "/*//label[text()='\" + label + "\"]";  
  expect(getBrowserFor(person).isVisible(_labelSelector)).toBe(true);  
});
```

```
function getBrowserFor(person) {  
  return (person === 'Both') ? browser : browsers[person];  
}
```

Learning Chimp

chimp.readme.io/docs

cucumber.io/docs/reference

jasmine.github.io

webdriver.io/api.html

Thank You!

Demo

- Medtech