Site Monitoring

Care what your customers care

Kenny Peng
kennyp@yahoo-inc.com
Aug 12, 2014
Why monitoring?

To ensure your service meets the requirements of your customers,

to satisfy and delight your customers,

to ensure your service meets the requirements of your customers.
Traditional website monitoring

- Apache active children
- Interface traffic
- CPU %
- Load average
- Used memory %
- TCP anomalies
But … web architecture is more and more complicated
But ... web architecture is more and more complicated
Server-level monitoring can’t detect …
How to prevent ? or be early to know?

Site Monitoring

Complete and continuous frontend testing
Concept of Site Monitoring

- Heartbeat rate monitoring
- Comprehensive Cost

Once per year
- Month
- Day

Full body Monitoring
- Comprehensive Cost

http://www.indushealthplus.com/wp-content/themes/twentyeleven/art_img/medical-testing.jpg
Concept of Site Monitoring

Server level monitoring

Comprehensive

Cost

Reusable all tests & run them continuously

Comprehensive

Cost

Why the cost should be $\downarrow$?

Reuse all existed page tests
Why are page tests hard to be reused?

- Only be used within push pipeline
- Only test specific test pages
- Hardcode test data
Design a reusable tests

Hardcode Configs:
Css selectors, css styles, html structures, … etc.

Accept input data:
Title, descriptions, language, currency, ads info, … etc.
Talk more about test data

*Tip: always get the most trustable data as test data*
Ex: hardcode configs

**hardcode css selectors**

```javascript
var mssLocators = {
    "image": ":root > div > h1",
    "title": ":root > div > h2"
};
```

**Assign different settings in different condition**

```javascript
self = this,
testParams = self.testParams,
yuiinstance,
MSSLocators = (testParams.monitoring) ? mss_gs_locators : mssLocators;
```
Ex: Accept input data

“title” is configurable

```javascript
44 +     {
45 +         "test": "test/mss_rendering.js",
46 +         "engine": "mocha",
47 +         "engineConfig": "./node_modules/td-test-utils/config/mocha-config.json",
48 +         "title": "${config.monitoring.td_applet_mss.title}",
49 +         "monitoring": true
50 +     }
```

Modify code to accept input data (hardcode originally)

```javascript
20     it('should display the correct MSS title', function (done) {
21         var check = yuiInstance.one(mssLocators.title).getHTML();
22         expect(check).to.equal('More from Fidelity');
23     });
24     var check = yuiInstance.one(MSSLocators.title).getHTML(),
25     title = (testParams.title) ? testParams.title : 'More from Fidelity';
26     +     expect(check).to.equal(title);
```
Tests composing

- Depends on different test frameworks/scripts
- Test efficiency (scenarios, browsers)
- Considering test results analysis and report
Ex: Test efficiency (merge test scenarios)

### Component 1 test scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Page</th>
<th>Test</th>
<th>Title</th>
<th>Description</th>
<th>Monitoring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$$\text{config.monitoring.url}$$</td>
<td><code>test/component_1.js</code></td>
<td>$$\text{config.monitoring.component_1.title}$$</td>
<td>$$\text{config.monitoring.component_1.description}$$</td>
<td>true</td>
</tr>
</tbody>
</table>

### Component 2 test scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Page</th>
<th>Test</th>
<th>Title</th>
<th>Description</th>
<th>Monitoring</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$$\text{config.monitoring.url}$$</td>
<td><code>test/component_2.js</code></td>
<td>$$\text{config.monitoring.component_2.title}$$</td>
<td>$$\text{config.monitoring.component_2.description}$$</td>
<td>true</td>
</tr>
</tbody>
</table>

---

Don’t need to load page twice

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Page</th>
<th>Test</th>
<th>Title</th>
<th>Description</th>
<th>Monitoring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$$\text{config.monitoring.url}$$</td>
<td><code>test/component_1.js</code></td>
<td>$$\text{config.monitoring.component_1.title}$$</td>
<td>$$\text{config.monitoring.component_1.description}$$</td>
<td>true</td>
</tr>
<tr>
<td>2</td>
<td>$$\text{config.monitoring.url}$$</td>
<td><code>test/component_2.js</code></td>
<td>$$\text{config.monitoring.component_2.title}$$</td>
<td>$$\text{config.monitoring.component_2.description}$$</td>
<td>true</td>
</tr>
</tbody>
</table>
Reports and Notification

- Reports classification (changes & test results)
- Importance of each test case/component is different
- Notification levels
Ex: difference importance

```json
......
{
    "id": "MediaContentFollow",
    "source_control": "git",
    "arrow_repo": "git@github.com:yahoo/arrow.git",
    "weight": 10,
    "importDescriptor": {
        "path": "/tests/arrow/descriptor.json",
        ..... 
    },
},
{
    "id": "gs_site_level_test",
    "source_control": "none",
    "arrow_repo": "none",
    "weight": 50,
    "importDescriptor": {
        "path": "/page-test-monitoring-descriptor.json",
        ..... 
    },
},
}```
Sponsored Program Monitoring Report

2014-07-22 06:07:11

Summary

|    | passed: 343 | failed: 12 | errors: 0 | ignored: 0 |

finance_driven

PASS

PASS

PASS

PASS

PASS

finance_hot-stock-minute

PASS

PASS
Design of Site Monitoring