Automating the Crossword

Testing Web Games at the New York Times

Phil Wells
@thephilwells
phil.wells@nytimes.com
October 2019
Games at the New York Times

● Team composition and processes
  ○ Cross-functional team building games for web, iOS, and Android
  ○ Agile teams combining test automation with manual QA processes

● What we’ve been up to
  ○ Supported and updated the online NYT Crossword
  ○ Released new games: Spelling Bee! Letter Boxed!

● Big theme for crossword improvement: Removing Shame for solvers
The Puzzles Get Tough

Doing Monday’s Puzzle

Doing Saturday’s Puzzle

Everyone is stupid except me

I glued my head to my shoulder.
Removing shame for puzzle solvers

- “Shame” is a big word
- Make solving enjoyable for Monday/Tuesday solvers
- Less Shame, More Game!

@thephilwells
How We Helped Solvers

Milestone Animations

Vertical Streaks

Autocheck
Software development is a puzzle, too

- A good outcome is possible
- The path is not straightforward
- We lose ourselves in the work
- It’s “our puzzle”
- When we can’t get it right, we might feel ashamed
Removing Shame In Software Development

- Tests are there to remove shame from software development
- Releasing code with bugs - **SHAME!**
- Tests that do more harm than good - **SHAME!**

Who you gonna call?

@thephilwells
Shame Busters!

Responsibility

Running Often

Really Quite Fast

Reliability

@thephilwells
Responsibility

Who owns quality on this team?

Everybody!
Without Shared Responsibility

- “I can’t read the test code.”
- “They can’t read my test code!”
- “I don’t even know what’s covered by our end-to-end tests.”
- “The tests are broken and the test engineer is on vacation. Guess it’ll just stay broken forever.”
“Web Developers Couldn’t Possibly Create Tests”
Who Needs To Read Test Code?

- Not just a Software Engineer in Test
- Not just the engineer who wrote it
- Every engineer on the team needs to be able to read the automated test code
You Can’t Test Everything End-to-End

- “Which Tests Should We Automate” - Angie Jones, SauceCon 2018
- Too much maintenance
- Too much time
- Too much noise
- Too much code!
On This Team, Everyone Writes Tests!

- Prioritized test coverage is part of our definition of done
- Features ship with unit and component tests
- Coverage gaps are handled in end-to-end tests using webdriver.io
Feature Milestones

- Test Strategy Kickoff
- Test Coverage Audit Before Launch
- Test Tech Debt Stories During “Code Hardening”
When Anyone Can Write A Test...

- Anyone can triage or fix a broken test
- Anyone can see what components the tests are checking
- Anyone can tell new team members how to do it

... anyone can read the tests!
BUSTED!
Run Often
And Report the Results

Roger that!
Infrequent, Unmeasured Tests

- “Whoa, the last time we ran tests on Staging was two months ago. There have been three releases in that time!”
- “I feel like there’s always a sprinkling of flakiness when the tests run, but I can’t pinpoint which tests might fail.”
- “My boss wants to know how much our test coverage has improved this quarter. What in G is lost to time? What in G am I, an archaeologist?”
Automate Your Automation

- Continuous Integration
- Sauce Labs
- Github
- Report Portal integration
How Often Do We Test?
In the beginning...

Developer creates a new branch off of develop

Developer makes a change

Developer pushes branch to Github

Unit Tests Run

End-to-end tests run in headless Chrome

Local Development

Docker Containers
Later...

Developer merges branch to develop

Unit tests run

End-to-end tests run on Saucelabs

QA team validates changes

Docker Containers

"Develop" Environment

"Develop" Environment
Finally...

Developer deploys code to Production

Unit tests run

Docker Container

@thephylwellis
Pretty Often!
Sauce Labs Analytics
BUSTED!
Reliability

Flakiness? Low fidelity?

Not on my watch!

@thephilwells
Unreliable Tests

● “Someone tainted these test accounts. Now I have to manually create a new group of test users.”
● “The API is down again. Guess I’ll go to lunch. Again.”
● “Our test pyramid looks more like a test waffle cone.”
Pushing Tests Down the Pyramid

- Couple UI testing with thorough unit test and service test coverage
- Focused UI tests
- Regularly de-dupe tests
Getting Rid Of Redundant Tests

- Bugs caught by UI tests should trigger the creation of unit and service tests
- Does this UI test provide any extra value?
- Don’t fall in love with your UI tests!
- “The Practical Test Pyramid” by Ham Vocke
  - https://martinfowler.com/articles/practical-test-pyramid.html
The right tools for the job!
User Management
User Management

- NY Times User Management Tool
- Easily Generate New Users
- Do Not Share Test Users
- Keep History Clean
Handling Race Conditions
Off to the races...

Problem: Trying to assert existence of short-lived elements on the page.

Case in point: milestone animations!
What Could Go Wrong?

Browser
- Create animation elements
- Destroy animation elements

Requests going over the network

WebDriver
- Fill in puzzle halfway
- Check for animation elements
Record Those DOM Events!

Browser
- Create Event Listener
- Create animation elements
- Event Listener Stores Events
- Destroy animation elements

WebDriver
- browser.execute()
- Fill in puzzle halfway
- Check for triggered events

@thephilwells
WARNING:
Here comes the code!
This “mutation” is an animation event!

Set up a watcher for these animation events

Set up a new array in the browser’s window object

Identify the web element where the event will occur

What kind of change events are we watching for?
const animationEvents = browser.execute( function () {
  return window.trackedEvents;
})

assert(animationEvents[0] && animationEvents[0].className).equals('animation-thing')
Using Mock Data
Why Use Mock Data?

- API development not complete
- Hard to get the real API to deliver broken data for edge cases
- Set up visual diff states
We wrote a library!

- Express-sidetrack middleware
- Delivers mock data to our application
- Internal for now, OSS some day
How It Works

Front End Requests → Mock Data Cookie? → Mock Data CDN

Back End Requests → Real API
Mock Data Gotchas

- Not a true end-to-end test
- Mock data have to be maintained
- Middleware has to be installed in the application under test
- Consider using an external proxy!
BU-SAT-ED!
Really Quite Fast Tests
Can’t We Fail Any Faster?
Yes we can!
Really Quite Slow Tests

- “Regression testing the app across all the supported platforms will take 2 weeks.”
- “I made a little change, but I can’t see if I broke anything without losing momentum.”
- “This single test takes a couple minutes to run on my local but times out after 5 minutes on Sauce Labs.”

@thephilwells
Disable Ads (most of the time)

- Slow to load
- Only explicitly tested in a few cases
- We can set a cookie to disable ads on test environments
Speeding Up Grid Interactions
Meet the Grid

- Made up of 144-225 cells
- Each cell contains 3 web elements
- Up to 675 individual web elements
- Some test cases require that all of the cells’ states be validated
## Traffic Jam

<table>
<thead>
<tr>
<th>Operation</th>
<th># of Network Calls</th>
<th>Seconds Network Traffic (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get all cells</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>For each cell, get answer text</td>
<td>255</td>
<td>22,500</td>
</tr>
<tr>
<td>Now try it on Sauce VM’s</td>
<td>255</td>
<td>225,000!!!</td>
</tr>
</tbody>
</table>

@thephilwells
Many Slow Requests

Browser

DOM

Web elements from DOM in browser

Webdriver.io

@thephilwells
Do It Locally With Cheerio

Browser

DOM

HTML

Webdriver.io

HTML

Cheerio

DOM

Web elements from DOM in local memory
Cheerio.js Example

https://github.com/thephilwells/wdio-cheerio-demo
How Are We Doing?

And What Work Remains
Measuring Our Progress

- Amount of rework
- Confident delivery
- Regression cycle length decreased, from days to hours
Support From Product Folks

- Making test automation part of our “Definition Of Done”
- Test effort considered during estimation
- The up-front cost of automation…

Embrace it!
Nobody’s Perfect

- Team members have different comfort levels writing end-to-end tests
- All features ship with unit/component tests, end-to-end tests sometimes need to catch up post-release
- Need to capture ROI
- Tests still flake out sometimes!
- *Software testing is a process of continuous shame elimination*
Questions?
There’s no shame in asking...

- https://www.linkedin.com/in/thephilwells/
- https://twitter.com/thephilwells
- https://github.com/thephilwells