

GoBusiness Sharing

GDS Quality Engineering Chapter

5th March 2020

Contents

1. Challenges We Faced

2. Overcoming Them

- a. The Social Challenge
- b. The Technical Challenge

3. Things We Learnt

Challenges

The Social Challenges

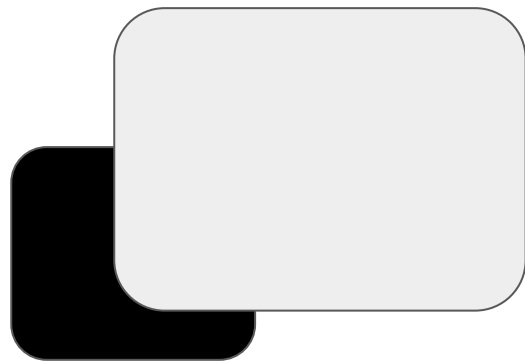
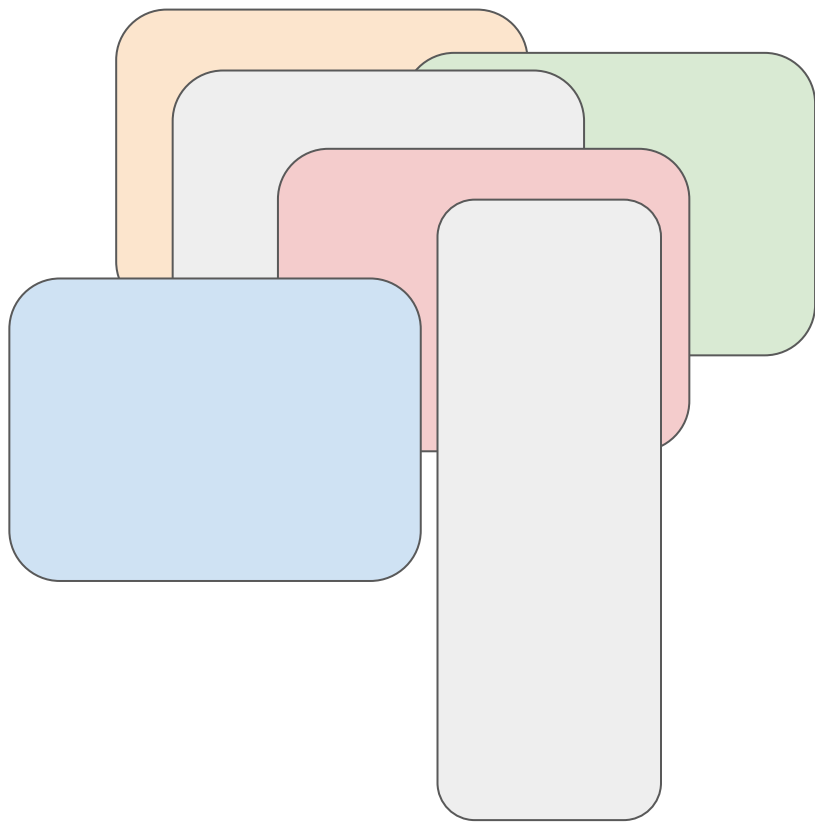
Siloed Roles



The Technical Challenges

Low Fidelity E2E





Challenges

A. Siloed Roles

- a. Slow Feedback
- b. Poor Quality

B. Patchy End To End

- a. Difficult to Maintain
- b. Low Fidelity

How We Overcame Them: Challenge 1

A Testing-Centric Approach

A Testing-Centric Approach

- A. Test Cases to be written before Development starts
- B. Developers to do self-testing before it's Ready For QA
- C. Anyone can pick up testing of features
- D. Testing is a team sport

Test Cases Before Development

Why?

- Allows us to realize what we don't understand about the feature
- Clarity on target

Everyone Tests

Why?

- Greater Ownership over the Product

How We Overcame Them: Challenge 2

Less Is More

Less is More

A. Deletion of Unstable Code

B. Flows Rather Than Functionality

Delete File



Are you sure you want to move this file to the Recycle Bin?



badtests.txt

Type: Text Document

Size: 0 bytes

Date modified: 11/28/2009 11:26 PM

Yes

No

Flows, not Functions

Why?

- A. Promotes Reusability
- B. Promote Readability, not Verbosity or Succinctness
 - a. DAMP not DRY

Data-PO-Flow

How?

- A. Repetition in Flows, not Functions
- B. Page Objects to mimic Front-End
 - a. promote Application Understanding
 - b. prevent duplicate code
- C. Have Best Practices for Standardization

Best Practices In Code

1. Strict Model
2. Choose a Convention to Prevent Astonishment

Less is More

A. Deletion of Unstable Code

B. Flows Rather Than Functionality

- a. Promoting Reusability
- b. Promoting Readability, not Verbosity or Succinctness
 - i. DAMP not DRY

C. Data - Page Object - Flow Structure

- a. Repetition in Flows, not Functions
- b. Page Objects to mimic Front-End, promote Application Understanding, prevent duplicate code
 - i. We can do reuse this way
- c. Have Best Practices for Standardization

Learnings

- A. Testing is a Team Effort
- B. Quality is a Team Effort
- C. Testing Pyramid
 - a. Unit, Integration and E2E tests should complement each other
 - b. E2E is expensive and kept to the necessary minimum
- D. Flow-based Functional Scripts

People want to reuse,
but don't know what
exists already

Good Quality comes from a
sense of ownership from the
team



